ACTION PC

SPÉCIAL OS/2

UN IBM A 9000 F

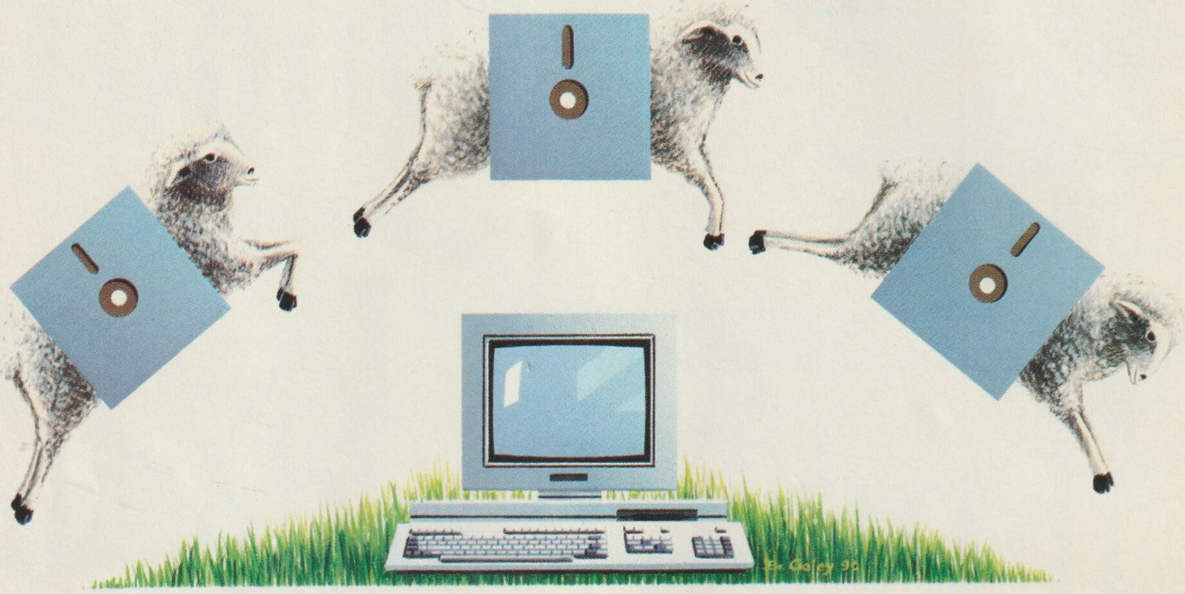


M 1131 - 6 - 25,00 F-RD

TOUT SUR LA HP-48SX

N°6 - MICRO-INFORMATIQUE - OCTOBRE/NOVEMBRE 1990 - 25 F

Si l'évolution de l'offre micro-informatique vous donne des insomnies,



faites le compte des solutions présentées au COMDEX®/Europe-SICOB MICRO

COMDEX®/Europe SICOB MICRO: L'événement européen de l'innovation et des solutions micro-informatique

3/6 OCTOBRE
Parc des expositions
PARIS-NORD

L'alliance du SICOB avec le COMDEX®, 1er Salon de la Micro aux USA vous permet de bénéficier d'une offre très étendue d'applications et de solutions MICRO-INFORMATIQUE adaptées à votre entreprise.

Un rendez-vous d'une importance capitale:

- 200 fabricants et éditeurs de logiciels internationaux dont une majorité d'américains présentent plus de 800 innovations technologiques
- Un espace "BANC d'ESSAIS" vous permet d'évaluer les performances de différentes gammes de produits
- Un point de rencontre destiné à faciliter vos échanges avec les professionnels et à optimiser votre visite
- Des tables rondes présentant les nouveautés guideront vos choix

Des moyens à votre disposition qui font du COMDEX®/EUROPE SICOB MICRO le lieu par excellence où se joue l'avenir de la micro-informatique professionnelle.

COMDEX Europe SICOB MICRO

LE SALON EUROPEEN
DE LA MICRO-INFORMATIQUE
PROFESSIONNELLE
EN AUTOMNE

Pour tout renseignement supplémentaire, contactez le Comité des Expositions de Paris Tél.: 33 (1) 49.09.60.92

Cerclez info pub 01

COMITE DES EXPOSITIONS DE PARIS

Ce salon est une des manifestations high-tech du Comité des Expositions de Paris

Deslegan

Sommaire

Actualités 10
Une rentrée très chargée : les nouveaux Compaq, le portable couleur de Sharp et le Poquet PC pour ne citer que quelques exemples.

L'aspect multitâche d'OS/2 42
Un dossier exceptionnel sur OS/2 réalisé par François Piette. La première partie explore les spécificités multitâches d'OS/2.

Programmer sous Présentation Manager 56
L'étude de la programmation sous PM constitue la seconde partie de notre dossier.

Exclusif sur OS/2 version 2.0 72
La troisième et dernière partie de notre dossier aborde la prochaine version 2.0 spécifique au processeur i386 d'OS/2.

Le calculateur HP-48 SX 76
Un dossier exclusif et exhaustif sur le nouveau calculateur scientifique de Hewlett-Packard et de ses possibilités de connexion avec un PC.

Hight-Tech : le bus EISA 90
L'extended ISA constitue le nouveau standard de l'industrie pour les bus à 32 bits. Action PC *fait le point sur cette nouvelle technologie.*

Initiation à la bureaucratique : PPP part 2 95
Le Paradox Personal Programmer et ses fonctions avancées, ou comment programmer sans programmer.

La communication 104
Dans un monde PC en continuelle évolution, il est parfois utile de faire le point. Première étape : la communication.

Turbo Forth 112
Un langage relativement peu connu et un produit français, étonnant !

Les logiciels Microsoft sur disque virtuel 116
Tous les logiciels récents de Microsoft pourront être installés sur disque virtuel avec cet unique programme conçu par Christophe Grosjean.

Qu'est-ce qu'un serveur de réseau ? 118

Le PS/1 d'IBM 122
Tout ce qu'il faut savoir sur le PS/1, nouvel ordinateur domestique d'IBM.

Les livres sur le Pascal 126
Tous les livres sur le Pascal examinés et analysés.

Le Shareware d'Action PC 136
Trois programmes très utiles, Kopy, Grep et VGATest.

Conquest of Camelot 156
Un jeu Sierra étonnant.

Les flashs logiciels 160
L'actualité du jeu PC.

ASC — ASIA STAR COMPUTER

☎ 43 87 36 03



CADRO/II 386 SX 16

Microprocesseur Intel i386 SX 16 MHz
- ◊ Boîtier AT 200 watts professionnelle
- ◊ 1 Mo RAM 80 ns, extensible à 2, 4, 8 Mo sur carte mère
- ◊ Lecteur 5"1/4 - 1,2 Mo, 2 Sorties séries + parallèle
- ◊ Contrôleur AT Bus "Rapide" pour 2 lecteurs et disque dur
- ◊ Clavier 102 touches Azerty ou Qwerty
- ◊ Microsoft Windows 3 français + MS-DOS 4.01 + Gw-Basic
- ◊ Livré avec 2 licences + 4 manuels + 10 disquettes

Modèle	Mode affichage	Disque Dur	Prix T.T.C
Cadro/II 40V	VGA couleur	45 Mo 28 ms	14 980 F
Cadro/II 40VS	Super VGA	45 Mo 28 ms	15 980 F
Cadro/II 80V	VGA couleur	84 Mo 24 ms	16 980 F
Cadro/II 80VS	Super VGA	84 Mo 24 ms	17 980 F



CADRO/IIi 386 SX 20

Microprocesseur Intel i386 SX 20 MHz
- ◊ Boîtier AT 200 watts professionnelle
- ◊ 1 Mo RAM 80 ns, extensible à 2, 4, 8 Mo sur carte mère
- ◊ Lecteur 5"1/4 - 1,2 Mo, 2 Sorties séries + parallèle
- ◊ Contrôleur AT Bus "Rapide" pour 2 lecteurs et disque dur
- ◊ Clavier 102 touches Azerty ou Qwerty
- ◊ Microsoft Windows 3 français + MS-DOS 4.01 + Gw-Basic
- ◊ Livré avec 2 licences + 4 manuels + 10 disquettes

Modèle	Mode affichage	Disque Dur	Prix T.T.C
Cadro/IIi 40V	VGA couleur	45 Mo 28 ms	16 180 F
Cadro/IIi 40VS	Super VGA	45 Mo 28 ms	17 180 F
Cadro/IIi 80V	VGA couleur	84 Mo 24 ms	18 180 F
Cadro/IIi 80VS	Super VGA	84 Mo 24 ms	19 180 F



CADRO/III 386 20

Microprocesseur Intel i386 20 MHz
- ◊ Boîtier AT 200 watts professionnelle
- ◊ 1 Mo RAM 80 ns, extensible à 2, 4, 8 Mo sur carte mère
- ◊ Lecteur 5"1/4 - 1,2 Mo, 2 Sorties séries + parallèle
- ◊ Contrôleur AT Bus "Rapide" pour 2 lecteurs et disque dur
- ◊ Clavier 102 touches Azerty ou Qwerty
- ◊ Microsoft Windows 3 français + MS-DOS 4.01 + Gw-Basic
- ◊ Livré avec 2 licences + 4 manuels + 10 disquettes

Modèle	Mode affichage	Disque Dur	Prix T.T.C
Cadro/III 40V	VGA couleur	45 Mo 28 ms	16 980 F
Cadro/III 40VS	Super VGA	45 Mo 28 ms	17 680 F
Cadro/III 80V	VGA couleur	84 Mo 24 ms	19 800 F
Cadro/II 80VS	Super VGA	84 Mo 24 ms	20 500 F



PRESIDENT/II 386 25

Microprocesseur Intel i386 25 MHz
- ◊ Boîtier AT 200 watts professionnelle
- ◊ 1 Mo RAM 80 ns, extensible à 2, 4 Mo sur carte mère
- ◊ Lecteur 5"1/4 - 1,2 Mo, 2 Sorties séries + parallèle
- ◊ Contrôleur SCSI pour 2 lecteurs et disque dur
- ◊ Clavier 102 touches Azerty ou Qwerty
- ◊ Microsoft Windows 3 français + MS-DOS 4.01 + Gw-Basic
- ◊ Livré avec 2 licences + 4 manuels + 10 disquettes

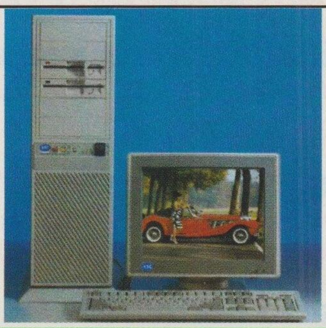
Modèle	Mode affichage	Disque Dur	Prix T.T.C
Président/II 80V	VGA couleur	84 Mo 24 ms	20 800 F
Président/II 80VS	Super VGA	84 Mo 24 ms	21 500 F
Président/II 120V	VGA couleur	120 Mo 20 ms	22 380 F
Président/II 120VS	Super VGA	120 Mo 20 ms	23 080 F

Nous sommes le premier distributeur sur Paris à vous livrer WINDOWS 3 en français sur nos configurations.

DISPONIBILITÉ IMMÉDIATE

LIVRÉ ET CONFIGURÉ SOUS EMBALLAGE D'ORIGINE

GARANTIE 1 AN PIECES & M.O.



PRESIDENT/III 386 33

Microprocesseur Intel i386 33 MHz (64 Ko antémémoire)
- ◊ Boîtier AT 230 watts professionnelle
- ◊ 4 Mo RAM 80 ns, extensible à 8 Mo sur carte mère
- ◊ Lecteur 5"1/4-1,2Mo+3"1/2-1,44 Mo, 2 Sorties séries + parallèle
- ◊ Contrôleur SCSI pour 2 lecteurs et disque dur
- ◊ Clavier 102 touches Azerty ou Qwerty
- ◊ Microsoft Windows 3 français + MS-DOS 4.01 + Gw-Basic
- ◊ Livré avec 2 licences + 4 manuels + 10 disquettes

Modèle	Mode affichage	Disque Dur	Prix T.T.C
Président/III 80V	VGA couleur	84 Mo 24 ms	30 830 F
Président/III 80VS	Super VGA	84 Mo 24 ms	31 530 F
Président/III 120V	VGA couleur	120 Mo 20 ms	32 280 F
Président/III 120VS	Super VGA	120 Mo 20 ms	32 980 F

ASIA STAR COMPUTER
28 Av. de Saint-Ouen
75018 PARIS
☎ 43 87 36 03 - Fax 40 08 00 74
Métro: La Fourche

OUVERTURE EN OCTOBRE
ASIA STAR COMPUTER GRENOBLE
57 Av. Marcel Cachin
38400 St Martin d'Hêres

OUVERTURE PROCHAINE
ASIA STAR COMPUTER LYON
22 Avenue Jean Jaures
68000 Lyon

Horaires d'ouverture du lundi au samedi de 9 h à 13 h - 14 h à 19 h - Accès facile par le Périphérique Nord, sortie Porte de St Ouen
Les clients qui désirent passer commande par téléphone pourront prendre livraison de leur configuration le lendemain sans aucune attente.

Photos non contractuelles - MS-DOS est une marque déposée de Microsoft

Publicité réalisée en PAO par PAONGRAPH® 42 77 60 90

SICOB
STAND 4A25



PHOTO NON CONTRACTUELLE

UNE GAMME COMPLETE :
8088-80286-80386-80386 SX-80486

MODEL aAT FAMILIO/II
80286-12 MHZ 1 MO RAM EXTENSIBLE A 4MO
1 LECTEUR 5"1/4 1.2 MO + DISQUE DUR 40 MO 28 MS
2 PORTS SERIE ET PARALLELE
8 SLOTS D'EXTENSION, CLAVIER AT 102 TOUCHES

FAMILIO/IIM	FAMILIO/IIV
MONITEUR MONOCHROME	MONITEUR COULEUR VGA
5860,00 F H.T. (6950,00 F T.T.C.)	8390,00 F H.T. (9950,00 F T.T.C.)

DISPONIBLE

CARTE VGA 1 Mo RAM — 1024x768 — 256 COULEURS — DRIVE WINDOWS 3

POINTS DE VENTE AGREES

★ PARIS REGION PARISIENNE ★

75 PARIS
38, rue du Chemin Vert
75011 PARIS

83, rue de Tolbiac
75013 PARIS
☎ 45 70 99 39
Fax 45 70 95 05

38, Av. de Versailles
75016 PARIS
☎ 42 30 87 36
Fax 42 24 41 61

62, rue des Entrepreneurs
75015 PARIS
☎ 45 78 26 58
Fax 45 77 82 74

28, Av. de St. Ouen
75018 PARIS
☎ 43 87 36 03
Fax 40 08 00 74

158, rue des Pyrénées
75020 PARIS
☎ 43 66 07 95
Fax 43 66 59 49

92 CLICHY
2, rue Sanzillon
92110 CLICHY
☎ 47 37 70 61
Fax 47 37 85 29

94 L'HAY LES ROSES
144, Av. de Stalingrad RN7
94240 L'HAY LES ROSES
☎ 46 75 99 33
Fax 46 87 40 33

94 SAINT MAUR
24, Av. de la Libération
94100 SAINT MAUR
☎ 48 86 42 99
Fax 43 97 09 12

★ PROVINCE ★

06 NICE
11Bis, Bd. Raimbaldi
06000 NICE
☎ 93 80 45 76
Fax 93 80 46 23

13 MARSEILLE
121, rue de l'Evêché
13002 MARSEILLE

29 BRETAGNE
"Gouvin"
29850 GOUESNOU
☎ 98 07 77 61
Fax 98 07 79 61

38 GRENOBLE
57, Av. Marcel Cachin
38400 ST.MARTIN D'HERES

45 ORLEANS
18, rue St. Paul
45000 ORLEANS
☎ 38 62 10 10
Fax 38 62 20 50

69 LYON
22, Av. Jean Jaurès
69000 LYON

LOUKIS (1) 46 66 06 38

Cerclez info pub 02

ILS NOUS FONT CONFIANCE

S.N.C.F. - ATOCHEM - ALCATEL - SLIGOS - COMPUTERLAND - RANDOM - BANQUE DE FRANCE - B.P.C. - CRÉDIT AGRICOLE - CRÉDIT DU NORD - BCEAO - BID - RTL - VALENTINE - FRANCE TELECOM - MINISTÈRE DE LA DÉFENSE - ÉDUCATION NATIONALE - CNRS - AFPA - INRA - UNESCO



Alif vous offre les **meilleurs prix** du marché

Alif vous **livre rapidement** sur toute la France

Alif **sélectionne** rigoureusement ses produits

Alif
(Nouvelle adresse 400 m²)
76, rue des Grands Champs
75020 PARIS
Tél. : 43.70.70.22
Fax : 43.70.71.66
(RER Nation ou M° Maraichers)

...DERNIÈRE MINUTE : IMPORTANTE BAISSE DE PRIX... DERNIERE MINUTE : IMPORTANTE BAISSE DE PRIX...

CONFIGURATIONS (moniteur inclus)	Disque dur	Mono type Hercules	Mono VGA	Couleur VGA	Couleur Multisync
ATurbo 286 PRO 1200 Boîtier Baby - CM 286/12 MHz 1 Mo RAM - Série // Lecteur 5¼ ou 3½ HD Clavier 102 touches	20 Mo	5490TTC	6290TTC	7590TTC	8690TTC
	40 Mo (28 ms)	5790TTC	6590TTC	7890TTC	8990TTC
	75 Mo (18 ms)	7590TTC	8490TTC	9690TTC	10890TTC
ATurbo 386 Pro 1600 SX idem sauf : Carte mère 386, 16 MHz SX	40 Mo (28 ms)	7290TTC	8090TTC	9390TTC	10490TTC
	75 Mo (18 ms)	9090TTC	9890TTC	11190TTC	12290TTC
	105 Mo (18 ms)	9790TTC	10590TTC	11890TTC	12990TTC
ATurbo 386 Pro 2000 idem sauf : Carte mère 386, 20 MHz	40 Mo (28 ms)	10490TTC	11410TTC	12810TTC	14080TTC
	75 Mo (18 ms)	12510TTC	13430TTC	14830TTC	16090TTC
	105 Mo (18 ms)	13310TTC	14230TTC	15630TTC	16890TTC
	160 Mo (18 ms)	17030TTC	17940TTC	19350TTC	20610TTC
ATurbo 386 Pro 2500 idem sauf : Carte mère 386, 25 MHz	40 Mo (28 ms)	11230TTC	12150TTC	13550TTC	14820TTC
	75 Mo (18 ms)	13250TTC	14170TTC	15170TTC	16830TTC
	105 Mo (18 ms)	14050TTC	14970TTC	16370TTC	17630TTC
	160 Mo (16 ms)	17770TTC	18680TTC	20090TTC	21350TTC
ATurbo 386 Pro 2500 + Boîtier vertical (6 emplacements 1/2 hauteur) - CM 386, 25 MHz + RAM cache 2 Ko RAM - Série // Lecteur 5¼ ou 3½ HD Clavier 102 touches	40 Mo (28 ms)	14300TTC	15200TTC	16600TTC	17850TTC
	75 Mo (18 ms)	16300TTC	17200TTC	18650TTC	19900TTC
	105 Mo (18 ms)	17100TTC	18000TTC	19450TTC	20800TTC
	160 Mo (16 ms)	20800TTC	21750TTC	23150TTC	24400TTC
	330 Mo (16 ms)	24950TTC	25900TTC	27300TTC	28600TTC
	660 Mo (16 ms)	30150TTC	31050TTC	32500TTC	33700TTC
ATurbo 386 Pro 3300 idem sauf : Carte mère 386, 33 MHz	40 Mo (28 ms)	17300TTC	18150TTC	19500TTC	20850TTC
	75 Mo (18 ms)	19250TTC	20200TTC	21600TTC	22850TTC
	105 Mo (18 ms)	20050TTC	20960TTC	22400TTC	23650TTC
	160 Mo (16 ms)	23800TTC	24700TTC	26100TTC	27350TTC
	330 Mo (16 ms)	27950TTC	28850TTC	30250TTC	31500TTC
	660 Mo (16 ms)	33100TTC	33990TTC	35400TTC	36650TTC
ATurbo 486 Pro 2500 idem sauf : Carte mère 486, 25 MHz + RAM cache	105 Mo (18 ms)	34100TTC	35000TTC	36400TTC	37700TTC
	160 Mo (16 ms)	37800TTC	38700TTC	40100TTC	41400TTC
	330 Mo (16 ms)	42000TTC	42900TTC	44300TTC	45500TTC
	660 Mo (16 ms)	47100TTC	48000TTC	49400TTC	50700TTC

OPTIONS : MS DOS (4.01) **890 F** -OS/2 : **N.C.** -Xenix : **N.C.** -2e lecteur **790 F** - 1 Mo sup. : **890 F** -Ext. 512 Ko VGA : **490 F** -Boîtier Mini tour : **490 F** -(Grand) : **990 F** -Souris + : **390 F** -(Pro) **690 F**
Co-processeurs : (287) : **1990 F** -(387 SX) : **2690 F** -(387-20) : **3390 F** -(387-25) : **4390 F** -387-33) : **4990 F**

CALCULEZ VOUS-MÊME !!!

Pour le prix d'une configuration de base chez nos « confrères », **Alif** vous offre :
- 1 assemblage de qualité et 72 h de tests,
- 2 Mo RAM (4Mo pour les 386-25+ et 386-33),
- 2 lecteurs : 5¼ 1,2 Mo + 3½ 1,44 Mo,
- Disque dur 40 Mo 28 ms (75 Mo 18 ms pour 386-25+ et 386-33),
- Carte VGA 16 bits + écran couleur VGA 14",
- 1 souris « Full » compatible Microsoft.

Votre configuration complète pour :
- 286, 12 MHz **9690F TTC seulement !**
- 386, 16 MHz (SX) **11190F TTC seulement !**
- 386, 20 MHz **14990F TTC seulement !**
- 386, 25 MHz **15390F TTC seulement !**

idem sauf
4 Mo RAM + Disque 75 Mo 18 ms + RAM cache
- 386, 25 MHz (cache) **21150F TTC seulement !**
- 386, 33 MHz (cache) **24100F TTC seulement !**

...PORTABLES... BAISSE DE TARIFS... PORTABLES... BAISSES...

LAPWORK 3140 :
286 - 16 MHz - 1 Mo RAM
Disque 40 Mo - Ecran EGA **16890F TTC**

LAPWORK 2100 :
idem sauf écran VGA **19690F TTC**

LAPWORK 8100 :
386 - 20 MHz - 2 Mo RAM
Disque 40 Mo - Ecran VGA **27790F TTC**
CADEAU : Sacoche + Pavé numérique



+ de 2500 produits micros aux meilleurs prix (TTC bien sûr)

Alif

« GONFLEZ » VOTRE PC

Cartes mères

XT 10 MHz	490 F
AT 286 / 12	1150 F
AT 386 / SX	2850 F
AT 386 / 20	5290 F
AT 386 / 25	6290 F

Avec mémoire cache

AT 386 / 25	6800 F
AT 386 / 33	9900 F
AT 486	N.C.

BOITIERS + ALIM.

XT / 150 W	740 F
Baby 200 W	790 F
Mini Tour / 200 W	1020 F
Big Tour / 230 W	1850 F
Big Tour / 300 W	3690 F
Alimentation	N.C.

DISQUES DURS DE 20 A 2600 Mo !!!

20 Mo	1550 F
30 Mo	1990 F
40 Mo	2470 F
80 Mo	4990 F
105 Mo	5490 F
160 Mo	7490 F
200 Mo	8490 F
330 Mo	11490 F
660 Mo	16790 F
710 Mo	19990 F
1300 Mo	75990 F
2600 Mo	152290 F

FILECARDS

20 Mo	2690 F
30 Mo	2890 F
40 Mo	3290 F

AUGMENTEZ VOTRE MÉMOIRE

Cartes RAM

XT 640 Ko	290 F
XT 6 Mo	1890 F
AT 2 Mo	790 F
AT 6 Mo	1890 F
AT 386, 6 Mo	1890 F
AT 386, 8 Mo	890 F
AT 386, 12 Mo	1290 F

COMPOSANTS RAM

4164-10	17 F
41256-10	20 F
41256-80	32 F
41464-10	36 F
44256-80	85 F
411000-10	75 F

SIM / SIP

(256 x 9)-80	250 F
(1 Mo x 9)-80	590 F

LECTEURS DISQUETTES

5¼ 1,2 Mo	590 F
3½ 1,44 Mo	590 F
Rack	95 F

CONTROLEURS

2 HD	450 F
2 FD/HD (1.2)	690 F
2 FD/HD (1.1)	850 F
Câbles	85 F
ESDI, SCSI, AT BUS	N.C.

COMMUNIQUEZ

Cartes Modem

PNB	1190 F
KX Tel 2	1750 F
KX 1200	5490 F

Cartes FAX

PNB Samantha	7990 F
Sofdit PC FAX	11290 F

ACCÉLÉREZ (x 5) L'EXPLOITATION DE VOS LOGICIELS CO-PROCESSEURS INTEL

8087	1410 F
80287	1960 F
80387/SX	2470 F
80387/20	3190 F
80387/25	4000 F
80387/33	4490 F

CARTES DIVERSES

Série 1 P	190 F
Série 2 P	250 F
Série // Jeu	270 F
2 séries // Jeu	390 F
Jeu	190 F
//	190 F
ARCNET	990 F
ETHER Net	1490 F

DIVERS

Souris OEM	195 F
Souris +	390 F
Souris PRO	690 F
Souris Microsoft	1420 F
Trackball	N.C.
Clavier/Trackball	790 F
Joystick	160 F
Clavier 102 touches	390 F
Scanner (Hain)	1690 F

FOURNITURES

Disquettes

5¼ 360 Ko	2,20 F
5¼ 1,2 Mo	6,90 F
3½ 720 Ko	6,90 F
3½ 1,44 Mo	17,90 F
Bac rangt 100 disq.	99,00 F
Listings rubans	
Cartouches streamers	

MONITEURS

12" Mono (Hercules)	790 F
14" Mono (Hercules)	990 F
14" Mono (VGA)	1190 F
14" Couleur (VGA)	2690 F
14" Couleur (Multisynch. 800 x 600)	3990 F
14" Couleur (Multisynch. 1024 x 768)	5350 F
(NEC 3D ou Sony)	5350 F
15" A4 Samsung	3990 F
NEC 2A/4D/5D, Idek	N.C.

CARTES ECRANS

MGP ou CGA	290 F
VGA 8 bits	750 F
VGA 16 bits (256 Ko)	850 F
VGA 16 bits Pro (256 Ko)	1390 F
VGA 16 bits (512)	995 F
VGA 16 bits Pro (512)	1890 F
NEC MGE	9990 F
Samsung A3/A4	N.C.
Paradise, Trident	N.C.

PROMO SPÉCIALE SICOB *(STOCKS LIMITÉS)*

IMPRIMANTES

CITIZEN 120 D 80 col. 9 aiguilles **1350F TTC**	**CITIZEN 124 D** 80 col. - 24 aiguilles **2490F TTC**	**CITIZEN Swift 24** 80 col. - 24 aiguilles Option couleur **2990F TTC**
DICONIX 150+ Jet d'encre Spéciale portable **3290F TTC**	**Canon BJ 130 E** Jet d'encre **4890F TTC**	**Hewlett Packard** DeskJet Jet d'encre **4490F TTC**
Canon PJ 1080 A Jet d'encre Couleur **5890F TTC**	**Canon LPB 4** Laser 4 P/mn **9490F TTC**	**Canon LPB 8** Laser 8 P/mn **15990F TTC**
Hewlett Packard Laserjet II P Laser 4 P/mn **9999F TTC**	**Hewlett Packard** Laserjet II D Laser 8 P/mn **22690F TTC**	**Hewlett Packard** HP Color Pro Traceur **9990F TTC**

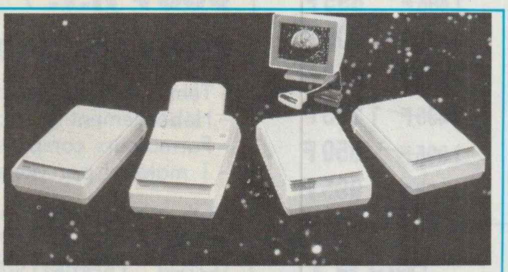
DISQUES DURS - ONDULEURS - STREAMER

Kalog (40 ms) 20 Mo MFM 30 Mo RLL **1550F TTC**	**Mitsubishi (28 ms)** 42 Mo MFM 65 Mo RLL **2290F TTC**	**Rodime (18 ms)** 75 Mo IDE **3890F TTC**
Wangtek Streamer 40/120 Mo **2990F TTC**	**Micropolis (16 ms)** 160 Mo ESDI **6490F TTC**	**ONDULEURS** 360 VA : 2690F TTC 550 VA : 3290F TTC 1000 VA : 5990F TTC

Moniteurs - Ext. Mémoires - Scanners - Logiciels

VGA Premium II 16 bits - 1024 x 768 +Drivers Windows 3 **1990F TTC**	**EMS Turbo 6 Mo** Drivers LIM. EMS 4.0 Ø Ko : 1890F TTC Le Mo : 690F TTC	**Windows 3** + Souris **1990F TTC**
Scanner à main 400 DPI **1690F TTC**	**Ecran VGA 14"** 640 x 480 **2490F TTC**	**Ecran Multisynchro 14"** 1024 x 768 **3990F TTC**
Umax UF 32 Scanner 300 DPI **14990F TTC**	**Umax UF 42** Scanner 400 DPI **17690F TTC**	**Umax UG 80** Scanner 300 DPI **22490F TTC**

SCANNERS PROFESSIONNELS UMAX



UF 32 (300 DPI) - UF 42 (400 DPI)

145 vrais niveaux de gris. 102 réglages de contraste et de lumière.
Fonctionne en mode textes, HALFTONES et MIXTES avec 32 fenêtres par page.
15 PATTERNES prédéfinies (dont une pour la télécopie CCITT) 10 secondes/page.
Modification de la résolution par le hardware 3 DPI par 3 DPI.
Livré avec sa carte RS 422, logiciel Mocroart (sous Windows).

UG 80 (300 DPI)

256 vrais niveaux de gris. **Seul scanner** à fonctionner en mode virtuel, capable de modifier une image 1 DPI par 1 DPI jusqu'à 800 %.
Fonctionne avec une interface SCSI (AT PS/2, APPLE).
Livré avec Picture Publisher (Windows), logiciel permettant de gérer les naunces de gris.
51 réglages de contraste - 51 réglage de lumière.
51 réglages d'intensité - 15 paternes prédéfinies.

TURBO OCR SCANNER *(Station de reconnaissance de caractères)*

Ensemble comprenant : Scanner UF 32 complet + Introducteur feuille à feuille + Omnipage 386 (considéré comme l'un des meilleurs logiciels de reconnaissance de caractères actuel).

22490F TTC seulement *(Fonctionne avec AT 386 + 4 Mo RAM)*

STATION OCR MAC

Ensemble comprenant : Scanner UG 80 complet + introducteur feuille à feuille + Omnipage Mac.

31900F TTC

(Toutes les marques citées sont déposées.)

Alif — MATERIEL LOGICIEL FORMATION MAINTENANCE

Tél. : 43.70.70.22 +
Fax : 43.70.71.66

ALIF - Dpt Micro - 76, rue des Grands-Champs - 75020 PARIS

BON DE COMMANDE *(à retourner à ALIF 20e)*

DESIGNATION	Qté	Prix Unit.	Total TTC
Frais de port : 43 70 70 22			
		Total TTC	

Société
Nom
Adresse
..................................
Tél. : Fax :

☐ Ci-joint mon règlement par chèque
☐ Je préfère régler en contre-remboursement

Cerclez info pub 03

MUSTINFO
INFORMATIQUE SERVICES

8, ave. du Garigliano
91600 Savigny-sur-Orge
Tél. : (1) 69 96 71 11

N°
expire le :

Monsieur :
Société :

AVEC LA CARTE "MUSTINFO"

LES AVANTAGES DE LA CARTE MUSTINFO

ASSISTANCE TÉLÉPHONIQUE
Tout adhérent en difficulté sur un matériel ou sur un logiciel pourra faire appel au Service Technique et sera ainsi assisté dans ses manipulations.

GARANTIE
Tous les matériels distribués par **Informatique Services** bénéficie de la garantie totale du constructeur, qui est généralement de 12 mois.

MAINTENANCE SUR SITE
Tout adhérent peut souscrire un contrat de maintenance sur site aussi bien pendant la période de garantie qu'après cette période. Un seul appel et le dépannage interviendra dans les plus brefs délais, en tous points du territoire Français Métropolitain.

FORMATION
La formation sur les produits les plus courants est organisée en séminaire de 4 à 6 personnes, suivant un calendrier mis à la disposition des adhérents, au prix de 900 Frs HT/jour. Possibilité de formation au sein de votre entreprise.

RAPIDITÉ DE LIVRAISON
La majorité des produits du catalogue est en stock et la livraison peut intervenir 48 heures après réception de commande.

PRODUITS TESTÉS
Seuls sont conservés au catalogue les produits et matériels de marque connue et d'une fiabilité contrôlée. Les matériels sont d'autre part, testés avant livraison.

FINANCEMENT ET PAIEMENT
– Les règlements pourront être effectués éventuellement par CARTE BLEUE.
– Un dossier de financement en crédit ou en leasing (crédit-bail) pourra être mis en place et un accord obtenu dans un délai de 24 heures.

...ET LES PRIX

SUPER-PROMOTIONS d'OCTOBRE POUR LES ADHÉRENTS

LOGICIELS

MICROSOFT	–35 %
LOTUS	–35 %
SAARI	–30 %
WORDPERFECT	–35 %

BORLAND

TURBO PASCAL 5.0	~~1 495 F~~	650 F
TURBO PASCAL 5.5	~~1 495 F~~	895 F
TURBO C 2.0	~~1 495 F~~	700 F
TURBO BASIC	~~995 F~~	450 F
QUATTRO Français	~~2 495 F~~	1 295 F
QUATTRO U.S.	~~1 995 F~~	650 F
REFLEX + WORKSHOP	~~1 995 F~~	995 F

Prix H.T.

POUR TOUT AUTRE LOGICIEL CONSULTEZ-NOUS

TOUTE COMMANDE D'ADHÉRENT DEVRA ÊTRE EFFECTUÉE PAR COURRIER ACCOMPAGNÉ DU RÈGLEMENT T.T.C. (TVA 18,6 %).
Ajouter **50 F** par logiciel, **100 F** pour petit matériel, **200 F** pour Tandon et portable pour **frais de PORT.**

FORMATION

900 F H.T. / par jour en séminaire

DANS UN CADRE AGRÉABLE

- Tennis, piscine, practice de golf,
- Hébergement possible,
- Formateurs compétents
- 1 matériel par personne

DOS - Initiation à l'informatique - DOS sous WINDOWS - WORD 4 et 5 - WORKS - EXCEL - LOTUS - FRAMEWORK III - D BASE (utilisateurs) - PAGEMAKER...

POUR TOUS RENSEIGNEMENTS **INFORMATIQUE SERVICES**

Tél. : 69 96 71 11
Fax : 69 05 32 00
Télex : 681 176

DEMANDE
– CALENDRIER COURS
– CATALOGUE GÉNÉRAL

A RETOURNER A : INFORMATIQUE SERVICES - Z.A.C. des Gatines
8-10, Av. du Garigliano - BP 31 - 91601 Savigny-sur-Orge Cedex

NOM DE LA SOCIÉTÉ : ____________________

NOM DU DEMANDEUR : ____________ Prénom : ____________ Profession ou Fonction : ____________

Adresse : ____________________

Code postal ☐☐☐☐☐ Ville : ____________ Pays : ____________

Téléphone : ____________

Désire recevoir ***GRATUITEMENT :***

☐ *LE CALENDRIER DES COURS ET LE TARIF.* ☐ *LE CATALOGUE GÉNÉRAL INFORMATIQUE SERVICES.*

INFORMATIQUE SERVICES

Vous assure le service et les prix

ET... VOUS BÉNÉFICIEREZ DE **SUPER PROMOTIONS PONCTUELLES...**

MONITEURS

NEC

MULTISYNC 2 A	~~5490 F~~	3490 F
MULTISYNC 3 D	~~6950 F~~	4850 F
MULTISYNC 4 D	~~13900 F~~	8390 F
MULTISYNC 5 D	~~25900 F~~	18450 F

EIZO

6500 paper white 21 pouces résolution 1664 x 1200	~~13000 F~~	9100 F
9060 S MULTISYNC couleur	~~7400 F~~	4890 F
9070 S MULTISYNC couleur 16 pouces	~~11000 F~~	7700 F

CARTES ECRAN

VGA 640 X 480	~~1950 F~~	980 F
VGA ETENDU 16 bits, 512 K, 1024 X 768	~~2950 F~~	1850 F
VIDEO SEVEN VGA 1024 i - 256 K	~~3500 F~~	2620 F
VGA 1024 i - 512 K, 1024 X 768	~~4500 F~~	3370 F

PARADISE, EIZO, ORCHID.

CARTES MODEM

KORTEX

KX TEL 2 + KX COM 2	~~1750 F~~	1100 F
KORTEX 1200 + KX COM 2	~~4990 F~~	3790 F
KORTEX 2400 + KX COM 2	~~7500 F~~	5590 F
LCE		nous consulter

ONDULEURS

ONDYNE 250 VA	~~2995 F~~	2190 F
ONDYNE 400 VA	~~5990 F~~	4190 F
ONDYNE 600 VA	~~7400 F~~	4950 F

IMPRIMANTES LASER

CANON		
LBP 4, 4 pages minute	~~13600 F~~	8690 F
LBP 8, 8 pages minute	~~20990 F~~	13990 F
LBP 8 III T	~~26900 F~~	18990 F
HEWLETT PACKARD		
LASER II P, 4 pages minute maintenance sur site comprise	~~12500 F~~	9790 F
LASER série III, 8 pages minute maintenance sur site comprise	~~19950 F~~	14950 F
KYOCERA		
F 800, 8 pages minute	~~21950 F~~	15690 F
F 1000, 10 pages minute	~~25950 F~~	18490 F
F 1200, 10 pages minute	~~39500 F~~	28390 F
EPSON		
EPL 7100, maintenance sur site comprise	~~12700 F~~	8690 F
TOSHIBA, PL 6, 6 pages minute installation et maintenance comp.	~~14950 F~~	9100 F
NEC POTSCRIPT		
SW 290, maintenance sur site comprise	~~31950 F~~	21950 F

SCANNERS

EPSON		
GT 1000 + EPSCAN II - Couleur et monochrome - format A7	~~8900 F~~	6230 F
GT 4000 + EPSCAN II - couleur et monochrome - format A4	~~17900 F~~	12590 F
GT 6000 + EPSCAN II - couleur et monochrome	~~20900 F~~	14630 F
CANON		
IX 30 F - monochrome	~~8800~~	6190 F
+ carte interface + image in	~~2100 F~~	1470 F
LOGITECH SCANMANN Plus	~~2850 F~~	1850 F
SCANMANN	~~1650 F~~	1050 F
HP		nous consulter

Tandon

TOSHIBA

EPSON

SAMSUNG

Canon

hp HEWLETT PACKARD

NOVELL

Prix H.T.

ADHÉSION à la carte "MUSTINFO"

A RETOURNER A : INFORMATIQUE SERVICES - Z.A.C. des Gatines
8 - 10, Av. du Garigliano - BP 31 - 91601 Savigny-sur-Orge Cedex

NOM DE LA SOCIÉTÉ : ____

NOM DU DEMANDEUR : ____ Prénom : ____ Profession ou Fonction : ____

Adresse : ____

Code postal ____ Ville : ____ Pays : ____

Téléphone : ____

Veuillez trouver ci-joint un chèque de **400 Frs** au titre de la cotisation, valable douze mois à compter de ce jour, libellé à l'ordre de : **INFORMATIQUE SERVICES**

Fait à : ____ le, ____ Signature : ____

CARTE A ÉTABLIR AU NOM DE : ____

Votre carte personnelle ou au nom de la Société, vous sera expédiée avec toutes les explications sur ses multiples avantages.

Cerclez info pub 04

International

PC Monthly

Belgium	200 B.Fr.
Denmark	48.00 D.Kr.
France	25 Fr.
Germany	11.20 D.M.
Great Britain	£ 2.80
Greece	680 Dr.
Italy	8000 Lire
Luxembourg	200 L.Fr.
Netherland	12.00 Fl.
Portugal	680 Esc.
Spain	660 Ptas
Switzerland	10.40 S.Fr.
USA	$ 5.00

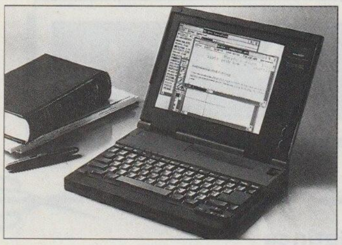
Belgium	200 B.Fr.
Denmark	48.00 D.Kr.
France	25 Fr.
Germany	11.20 D.M.
Great Britain	£ 2.80
Greece	680 Dr.
Italy	8000 Lire
Luxembourg	200 L.Fr.
Netherland	12.00 Fl.
Portugal	680 Esc.
Spain	660 Ptas
Switzerland	10.40 S.Fr.
USA	$ 5.00

Supplément d'Action PC N°4 — Depuis 1989

ACTUALITES MICRO

LES PORTABLES PRENNENT DES COULEURS

Sharp crée l'événement en commercialisant le premier compatible portable doté d'un écran couleur. En effet, le PC-8041, doté d'un processeur i386 cadencé à 20 MHz, reçoit un écran couleur à cristaux liquides de 14", compatible VGA (résolution 640 x 480 points et affichage de 16 couleurs parmi 512) et MCGA (320 x 200 points et 256 couleurs affichées parmi une palette de 512). Un essai du PC-8041 dira si les problèmes inhérents à ce type d'affichage (vision décalée et perturbée, couleurs « baveuses ») ont été résolus.



Le Sharp PC 6220, un challenger sur le marché des « Notebook ».

Les performances et l'équipement du PC-8041 sont dignes d'un système portable de haut de gamme. La mémoire vive de 2 Mo peut être portée à 8 Mo, par addition de modules de 1 Mo. Elle est en outre compatible avec la norme LIM EMS 4.0. Ce modèle est équipé d'un disque dur de 40 Mo (temps d'accès 19 ms) et une unité de disquettes 3"1/2 de 1,44 Mo. Le clavier de 94 touches comporte un pavé numérique séparé. L'utilisateur dispose d'une interface série RS-232, d'un port parallèle, d'un port souris, d'une prise pour un clavier de 102 touches externe, d'une prise pour un moniteur VGA analogique externe, et enfin d'un port pour une unité de disquettes 5"1/4 supplémentaire. Notons également deux connecteurs d'extension AT de 16 bits, l'un pour une carte longue, l'autre pour carte demi-longueur.

Le Sharp PC-8041 est livré avec le MS-DOS 4.01. Il pèse 13,5 kg et coûte 64 900 FHT.



L'événement : un écran LCD VGA en couleur.

D'autre part, Sharp annonce la disponibilité de son modèle PC 6220 de type « Notebook ». D'une taille légèrement inférieure au format A4 (21 x 29,7 cm), il pèse 2 kg. Le PC 6220 intègre un processeur 80286 à 12 MHz. La seule unité de stockage est un disque dur de 20 Mo — au format 2"1/2 — et disposant d'un temps d'accès de 23 ms. La mémoire vive de 1 Mo est extensible à 3 Mo directement sur la carte mère. L'écran à cristaux liquides VGA de 10 pouces assure une résolution de 640 x 480 points, avec 16 niveaux de gris. Sharp livre le PC 6220 avec le DOS 4.01, un utilitaire d'autodiagnostic, et Laplink 2.11, logiciel de transfert. Un emplacement permet d'installer un modem spécifique.

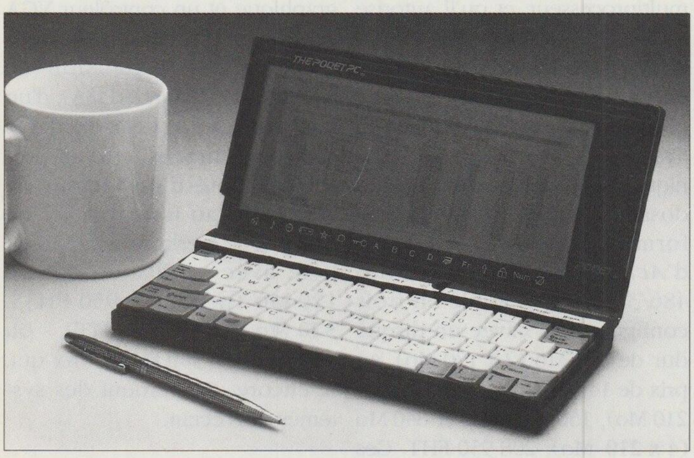
Le PC 6220 fonctionne sur le secteur, grâce à un adaptateur fourni, ou sur batterie, lui assurant environ 3 heures d'autonomie. Le PC 6220 est commercialisé dès à présent au prix de 27 950 FHT. Un lecteur de disquettes externe de 3"1/2 peut être adjoint au PC 6220, de même qu'un pavé numérique externe, et un boîtier d'extension offrant des connecteurs supplémentaires.

Service lecteur n° 1

LE POQET PC DEBARQUE !

Près d'un an après sa présentation officielle, dans le cadre du Comdex Fall 1989, le Poqet PC apparaît sur le marché français. Ce micro-ordinateur portatif autonome se distingue par des « mensurations » remarquables : 220 x 100 x 25 mm et seulement 450 grammes, et une alimentation assurée par deux piles alcalines AA qui lui confèrent une centaine d'heures d'autonomie !

Données étonnantes, d'autant que le Poqet PC est un véritable compatible. Il utilise un processeur 80C88 cadencé à 7 MHz, à faible consommation d'énergie. Le Poqet PC reçoit un écran monochrome à cristaux liquides, de 80 caractères et 25 lignes, compatible avec les modes MDA et CGA (résolution maximum de 640 x 200 points). Le clavier utilisé compte 77 touches, dont 12 touches de fonctions.



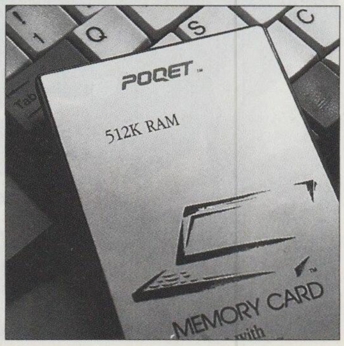
450 g seulement mais un écran de 25 lignes et 80 colonnes !

Le Poqet PC est livré avec 512 Ko de mémoire vive en standard, extensible jusqu'à 1,536 Mo par addition de deux cartes RAM. La mémoire du Poqet PC repose en effet sur l'utilisation de « cartes mémoires Poqet », d'une capacité de 512 Ko, qui prennent place dans les connecteurs prévus à cet effet. Les progrès attendus dans ce domaine laissent espérer une capacité de 2 Mo par carte, et ce dans un proche avenir — Poqet évoque le début de l'année 1991.

Les cartes peuvent comporter de la mémoire vive supplémentaire, ou encore des logiciels adaptés aux exigences du format. Un certain nombre d'éditeurs commercialise des versions Poqet de leurs produits : Lotus, avec Lotus 1-2-3, Metro, et Agenda, WordPerfect, avec WordPerfect Executive et WordPerfect 5.0, et encore Alpha Software (Alpha Works), ou Xyquest (Xywrite III Plus).

Néanmoins, le nombre réduit de logiciels disponibles a conduit Poqet à fournir son PC avec cinq programmes : PoqetNote, un traitement de textes simplifié, PoqetAgenda, agenda électronique, PoqetAdresses, répertoire téléphonique, PoqetCalculatrice, et PoqetDialogue, logiciel d'émulation de terminal. Le Poqet PC abrite également PoqetLink, un utilitaire de transfert de fichiers. La connexion avec un PC de bureau s'opérant grâce au câble de liaison, via la prise RS-232. Le Poqet PC est livré avec le MS-DOS 3.3 et le GW Basic. Sa distribution débutera au début du mois d'octobre. Le Poqet PC sera alors proposé au prix de 13 500 FHT. Suivront un certain nombre de périphériques, dont une unité de disquettes portable, et des câbles de liaison pour imprimantes et modems.

Le Poqet PC peut recevoir deux cartes d'extension mémoire de 512 Ko chacune)



Service lecteur n° 2

UN PORTABLE A DISQUE DUR AMOVIBLE



Le Rein M5 reçoit un disque dur amovible de 20,40, ou 120 Mo.

Par rapport à un classique portable autonome, le Rein M5 présente la particularité de disposer d'un disque dur amovible, choisi parmi les trois modèles disponibles, d'une capacité de 20, 40, ou 120 Mo. Il s'agit d'options, venant accroître la puissance du M5.

Ce portatif, utilisant un processeur 80286 à 12 MHz, dispose en standard de 1 Mo de mémoire vive, extensible à 2 Mo, d'un lecteur de disquettes interne de 3"1/2, et d'un clavier de 86 touches. Il comporte également un écran monochrome à cristaux liquides, rétro-éclairé, compatible MDA, CGA, et EGA (résolution maximum de 640 x 480 points). Le Rein M5 fonctionne sur le secteur, ou de manière autonome grâce à une batterie NiCad, qui lui confère quatre heures d'autonomie. Le M5 pèse 5,2 kg sans sa batterie, et 7,4 kg avec.

La panoplie de base du M5 comprend MS-DOS 4.01 et le GW Basic. Son prix est alors de 27 800 FHT, auquel s'ajoute éventuellement le coût d'un disque dur, soit 5 000 FHT pour le 20 Mo, ou 7 000 FHT pour le 40 Mo. La version de 120 Mo ne sera pas disponible avant le mois de novembre.

La société ASAP commercialise une large gamme de périphériques auxiliaires : extension mémoire de 1 Mo (4 000 FHT), lecteur de disquettes 5"1/4 externe (3 000 FHT), batterie supplémentaire (2 750 FHT), etc.

Le Rein M5 dispose d'un port parallèle, d'une interface série, d'une prise RGB (pour y connecter un moniteur couleur EGA), et un port pour clavier de 102 touches. Enfin, le M5 bénéficie d'un connecteur pour carte AT au format 2/3.

Service lecteur n° 3

LE SOUFFLE DU MINISPORT

Le MinisPORT HD de Zenith Data Systems donne la priorité à la portabilité. S'il se contente d'un processeur 80C88 à 10 MHz, il offre des dimensions (315 x 249 x 33 mm) et un poids (2,7 kg avec la batterie) qui facilite son transport.

caractères. Fonctionnant en mode CGA, il assure une résolution de 640 x 200 points avec huit niveaux de gris.

L'alimentation fait appel à une batterie au nickel cadmium, assurant environ trois heures d'autonomie, ou à un adaptateur secteur



Le MinisPORT : 2,7 kg et un disque dur de 20 Mo.

Disposant de 1 Mo de mémoire vive, le MinisPORT HD intègre un disque dur de 20 Mo (temps d'accès de 23 ms), et se passe de lecteur de disquettes interne. Un connecteur dédié permet néanmoins d'y raccorder un lecteur au format 3"1/2 ou 5"1/4.

Outre le BIOS habituel, la mémoire morte du MinisPORT HD contient la version 3.3 du MS-DOS, le GW Basic, ainsi qu'un programme utilitaire de transfert de fichiers, FastLynx LX. Ce logiciel permet d'effectuer des échanges de données avec une machine de bureau, par l'intermédiaire du câble de liaison fourni avec le MinisPORT HD.

Le clavier de 80 touches comporte 12 touches de fonctions séparées. L'affichage est confié à un écran à cristaux liquides inclinable, de 25 lignes de 80

110/220 volts. Outre le recours à des technologies économes en énergie (CMOS pour le processeur en particulier), divers dispositifs classiques permettent de réduire la consommation d'énergie : mise en veille automatique de l'écran et du disque dur après une durée d'inactivité déterminée, etc. L'arrière du MinisPORT abrite une interface série, un port parallèle bidirectionnel, une prise RVB pour moniteur analogique, ainsi que le connecteur à 20 broches destiné au branchement du lecteur de disquettes externe. Le MinisPORT HD peut également recevoir une carte modem.

Zenith distribue deux versions de son MinisPORT HD, avec ou sans lecteur de disquettes 3"1/2 externe, au prix de, respectivement, 18 450 FHT et 16 950 FHT.

Service lecteur n° 4

COMPAQ : DITES 33

Alors que certains constructeurs se battent sur le marché des PC bon marché, Compaq porte résolument ses efforts dans le secteur des systèmes de très haut de gamme. Cette stratégie se traduit aujourd'hui par la présentation de trois nouveaux



Le Systempro se dote du processeur i486 à 33 MHz.

micro-ordinateurs dotés de processeurs cadencés à 33 MHz.

Ainsi, le Systempro peut dorénavant être équipé de la version à 33 MHz du i486. Il dispose de 512 Ko de mémoire cache externe et de 8 Mo de mémoire vive en standard (capacité pouvant être portée à 256 Mo). Rappelons que le Systempro est un système multiprocesseur, et qu'il autorise toutes les combinaisons imaginables : 386 + 386, 386 + 486, ou encore 486 + 486. Pour plus de précisions sur la philosophie du Systempro, et un exposé technique complet, reportez-vous au dossier consacré à cette plate-forme dans le numéro 6 d'*Action PC*. Le Systempro 486/33 est disponible en trois configurations : avec un disque dur de 240 Mo (2 x 120 Mo), au prix de 145 950 FHT, 420 Mo (2 x 210 Mo), 166 950 FHT, et 840 Mo (4 x 210 Mo), 208 950 FHT. Ces prix s'entendent pour des systèmes complets, sans moniteur toutefois. Compaq propose également une carte processeur i486 à 33 MHz, pouvant remplacer la carte 386/33 des Systempro 386/33, ou être utilisée comme seconde carte processeur, au prix de 48 000 FHT.

La famille des Deskpro accueille, quant à elle, un modèle équipé du même processeur Intel i486 à 33 MHz. Le Deskpro 486/33L dispose de 4 Mo de mémoire vive en standard (extensible à 100 Mo), une carte graphique et un contrôleur VGA accéléré. Il dispose de sept connecteurs d'extension de 32 bits EISA (Extended ISA), d'un port parallèle, de deux interfaces série asynchrones, et d'un port souris. Doté d'un lecteur de disquettes au format 3"1/2 de 1,44 Mo, le Deskpro 486/33L est livré avec un disque dur de 120 Mo (prix : 97 950 FHT), 320 Mo (115 950 FHT), ou 650 Mo (136 950 FHT). Prix qui, là encore, concernent des systèmes sans écran.

Dernière nouveauté, le Deskpro 386/33L intègre un processeur 386 cadencé à 33 MHz. Ce modèle dispose de 64 Ko de mémoire cache et de 4 Mo de mémoire vive en standard, extensible à 100 Mo

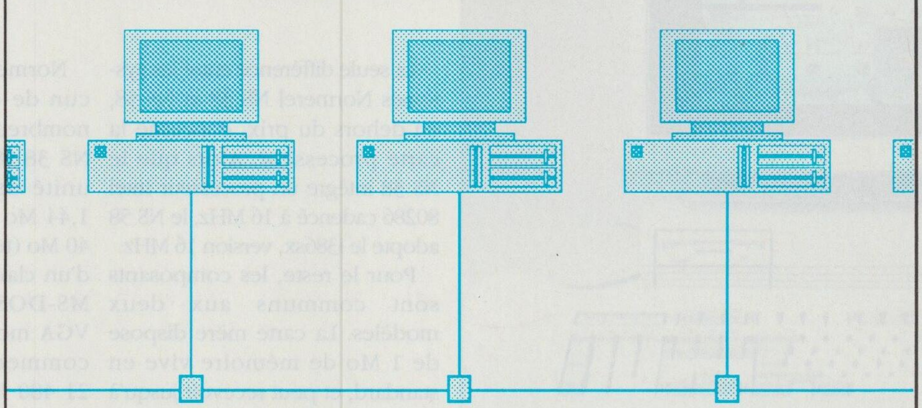
Le Compaq Deskpro 486/33L en application CAO.



Pourquoi faire compliqué quand on peut faire si simple ?

Réseaux locaux RPTI-NET©

Le meilleur rapport performances / prix du marché



RPTI-NET est une marque déposée de RPT Intergroups, PC/XT, PC/AT et PS/2 d'IBM et MS-DOS de Microsoft.

- ✔❑ Compatible PC/XT©, PC/AT©
- ✔❑ Compatible AT/386, PS/2©
- ✔❑ Compatible MS-DOS© 3.x et 4.x
- ✔❑ Compatible NOVELL
- ✔❑ De 1 Mbits/sec. à 10 Mbits/sec.
- ✔❑ Simple à installer
- ✔❑ Simple à utiliser
- ✔❑ Partage de disques
- ✔❑ Partage d'imprimantes
- ✔❑ Partage de logiciels
- ✔❑ Pas de serveur dédié
- ✔❑ Peu d'occupation mémoire
- ✔❑ Jusqu'à 255 postes

Importateur - Distributeur

H.A. Informatique S.A. / Galeries de Paname / 67 bld. Richard Lenoir / 75011 PARIS / Téléphone (1) 43 38 91 39 / Fax (1) 43 38 27 10

Cerclez info pub 05

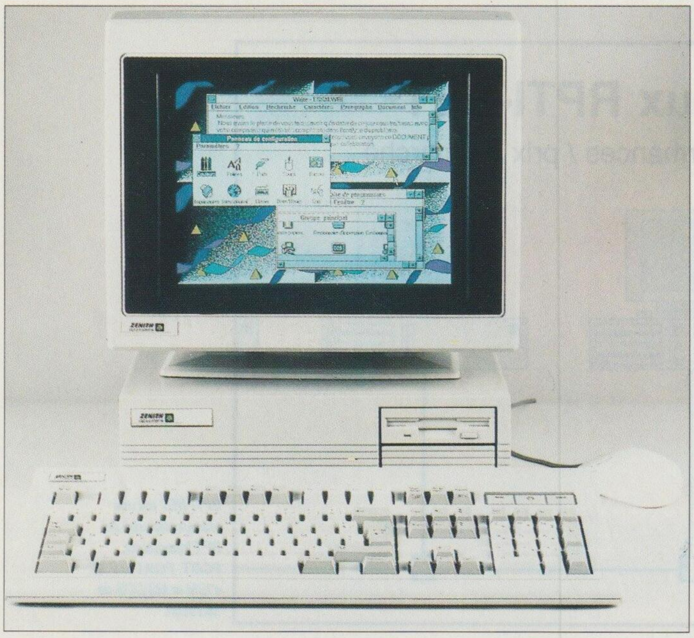
(contre 16 Mo pour le modèle 386/33 actuel).

Il dispose également de sept connecteurs de 32 bits EISA, alors que le 386/33 de base se contente de connecteurs ISA. Le 386/33L intègre enfin un contrôleur de disque dur ESDI, qui améliore d'environ 25 % les performances du disque dur (données constructeurs). Les prix du Deskpro 386/33L, sans écran, sont de 68 950 FHT pour la version munie du disque dur de 120 Mo, 86 950 FHT (320 Mo) et 107 950 FHT (650 Mo).

L'arrivée de ces systèmes entraîne une baisse des tarifs des modèles Systempro 386/33 — d'environ 15 % —, Deskpro 486/25 (- 20 %), 386/33, et 386s.

Service lecteur n°5

LE CONCEPT MODULAIRE AU ZENITH



286 ou 386sx par simple substitution de la carte processeur.

Simple 286 à l'origine, le Zenith Z-286 LP peut se muer en un 386sx par substitution de la carte processeur. Cette opération, dont le constructeur assure qu'elle ne prend que quelques minutes, entraîne un surcoût de 3280 FHT, prix de la carte d'extension 386sx.

Mis à part l'échange du processeur, 386sx à 16 MHz contre 80286 cadencé à 12 MHz, les caractéristiques de la machine demeurent inchangées. Le Z-286 LP intègre un contrôleur graphique VGA et dispose de 1 Mo de mémoire vive (extensible à 16 Mo). Il comporte un lecteur de disquettes au format 3"1/2 de 1,44 Mo, et offre trois connecteurs d'extension. Les prix dépendent directement du modèle de disque dur retenu, 10 Mo (temps d'accès de 28 ms) — prix du système sans écran 15 450 FHT — ou 40 Mo (28 ms) — prix 17 950 FHT. Les tarifs incluent le MS-DOS 3.3, la version 2.11 de Microsoft Windows, ainsi qu'une souris compatible Microsoft.

Service lecteur n°6

STATIONS DE RESEAUX : LA SAISON DES TULIP

Après les modèles 286, Tulip dévoile une station de travail pour réseaux locaux basée sur un processeur Intel 386sx à 16 MHz. Dénué de disque dur, le Tulip WS 386SX est proposé à partir de 16 600 FHT, prix de la configuration de base avec écran monochrome mais dénuée de lecteur de disquettes. Il est équipé de 2 Mo de mémoire vive, pouvant être étendue à 4 Mo directement sur la carte, et jusqu'à 16 Mo par ajout d'une carte mémoire. Le système gère la mémoire paginée LIM EMS 4.0. Le WS 386SX possède de surcroît un système de protection par mot de passe et un dispositif de verrouillage clavier. L'utilisateur dispose de trois connecteurs d'extension pleine longueur ISA de 16 bits, d'une prise parallèle, d'une interface RS-232, et d'un port souris. Le Supervisor Workstation Kit permet d'installer et de configurer la station WS 386SX.

Service lecteur n°7



Tulip WS 386SX, un 386sx s'épanouit.

SEMBLABLES ET SI DIFFERENTS

La seule différence entre les systèmes Normerel NS 38 et NS 58, en dehors du prix, concerne la carte processeur. Alors que le NS 38 intègre un processeur Intel 80286 cadencé à 16 MHz, le NS 58 adopte le i386sx, version 16 MHz.

Pour le reste, les composants sont communs aux deux modèles. La carte mère dispose de 1 Mo de mémoire vive en standard, et peut recevoir jusqu'à 5 Mo par addition de barrettes SIMM. Les systèmes abritent trois connecteurs d'extension AT de 16 bits, un connecteur 8 bits, une interface série, et un port parallèle. Sans oublier la carte graphique VGA, livrée en standard.

Normerel propose pour chacun de ces deux modèles de nombreuses configurations. Le NS 38 (80286), équipé d'une unité de disquettes 3"1/2 de 1,44 Mo et d'un disque dur de 40 Mo (temps d'accès de 28 ms), d'un clavier de 102 touches, du MS-DOS 4.01, et d'un écran VGA monochrome de 14", est commercialisé au prix de 21 480 FHT. Doté d'un disque dur de 100 Mo (19 ms), son prix passe à 25 480 FHT. Dans des configurations similaires, le NS 58 coûte quant à lui respectivement 24 980 FHT (disque dur 40 Mo) et 28 980 FHT (100 Mo).

Service lecteur n°8

Un seul boîtier pour deux « motorisations », 80286 ou i386sx.



Un logiciel de comptabilité gratuit*, c'est déjà un problème de comptabilité en moins.

Hypercut

COTE OUEST présente Bilan Plus version 2.0

En informatique, il y a rarement de miracles. Rien ne remplacera jamais un véritable essai "grandeur nature".

Chez COTE OUEST, nous vous proposons d'essayer gratuitement la dernière version de notre comptabilité Bilan Plus. De la saisie des écritures à l'édition complète du bilan, vous pourrez tout tester à votre rythme et découvrir tranquillement la simplicité, la rapidité et l'efficacité de Bilan Plus.

Sur simple appel, vous profiterez des services d'un conseiller COTE OUEST pour une évaluation vérité de vos besoins.

Pour obtenir le logiciel d'évaluation Bilan Plus ou recevoir la liste de nos revendeurs-conseils, retournez le coupon-réponse à :
COTE OUEST - B.P. 291 - 44010 Nantes cedex 01 ou téléphonez au 40 73 20 21 à Nantes.



Bilan Plus version 2.0

*Logiciel d'évaluation compatible avec la version complète.

Cerclez info pub 06

Je désire recevoir, gratuitement et sans engagement de ma part :
☐ Le logiciel d'évaluation Bilan Plus version 2.0
(merci de préciser le format de votre lecteur : ☐ 5 ¼ ☐ 3 ½)
☐ La liste de revendeurs-conseils COTE OUEST.

M. ______________ Société ______________
Adresse ______________
Code postal ______________ Ville ______________ Tél. ______________

COTE Ouest SOFT EDITIONS

UN 386 A 20 MHZ CHEZ PHILIPS

Le modèle P3350, présenté au début du mois juillet, prend place au milieu de la gamme 386 de Philips. Il intègre un processeur Intel i386 cadencé à 20 MHz.



80386 à 20 MHz et un disque dur de 100 ou 200 Mo.

Il propose en standard 1 Mo de mémoire vive, extensible à 8 Mo, un lecteur de disquettes 3"1/2 de 1,44 Mo, une carte et un contrôleur graphique compatible VGA. Il peut être doté d'un disque dur de 100 Mo ou 200 Mo (temps d'accès 19 ms). Le système P3350 comporte deux emplacements en façade destinés à recevoir des unités de sauvegarde supplémentaires, disque dur, CD-ROM, bande, etc. L'utilisateur dispose également de cinq connecteurs d'extension pour cartes 16 bits pleine longueur. La version 100 Mo du Philips P3350 est commercialisée depuis le mois de juillet, au prix de 35 500 FHT , sans moniteur.

Service lecteur n°9



Le système ETV 4000 S d'Olivetti.

UN PC TENDANCE SECRETARIAT

Le système ETV 4000 S d'Olivetti est tout à la fois un micro-ordinateur compatible et un poste de travail complet pour les secrétariats.

Particulièrement compact, l'ETV 4000 S intègre un cœur de PC — basé sur un processeur 80286 à 12 MHz —, un écran VGA monochrome ou couleur, ainsi qu'une imprimante graphique à transfert thermique. De plus, ce système est livré avec la version 2.11 de Microsoft Windows, et un logiciel de bureautique sous Windows. Aux classiques fonctions de traitement de textes, ce programme ajoute des fonctions d'importation et de gestion de graphiques, un utilitaire de capture d'images numérisées. Une simple touche permet de transformer l'ETV 4000 S en une banale machine à écrire, option, paraît-il, fortement appréciée des secrétaires.

L'ETV 4000 S est disponible au prix de 30 900 FHT pour la version monochrome, et 32 900 FHT en configuration couleur.

Quelques précisions sur les caractéristiques techniques de ce système s'imposent. Il dispose de 1 Mo de mémoire vive, capacité pouvant être portée à 2 Mo. Concernant l'archivage, l'ETV 4000 S s'appuie sur un disque dur de 20 Mo (temps d'accès de 27 ms) et un lecteur de disquettes au format 3"1/2 de 1,44 Mo. Le moniteur VGA de 14" est doté d'un écran plat. Sa résolution maximale atteint 640 x 480 points. Il est disponible en versions monochrome et couleur. Une souris compatible Microsoft est livrée en standard. Olivetti propose en option, un boîtier d'extension pour deux cartes AT pleine longueur, un bac d'alimentation automatique feuille à feuille, un mécanisme d'alimentation de papier en continu, un support d'écran, et un logiciel de vérification orthographique.

Service lecteur n°10

X-WINDOWS : LE RISC MOINS CHER

Les processeurs RISC, après avoir envahi les stations de travail, s'attaquent désormais aux terminaux graphiques. Le nouveau terminal X-Windows de Samsung (le SGS-19) sera pourvu du processeur RISC de 32 bits Am29000 d'AMD. Bien qu'équipé de la version la moins rapide de l'Am29000 (16 MHz) ce terminal aurait néanmoins une vitesse quatre fois supérieure à celle de ses concurrents par le prix. L'Am29000 pilotera un affichage monochrome de 48 cm, avec une résolution de 1280 x 1024 points. Associé à 256 Ko mémoire vive (RAM) vidéo et 2 Mo de mémoire vive dynamique d'instructions et de données, il assurera toutes les fonctions, y compris graphiques du SGS-19. Ce terminal est le premier d'une gamme à base de processeurs RISC.

Service lecteur n°11

■ La série M de Victor a été retenue par HEC pour équiper les étudiants, les enseignants, et les services administratifs de la prestigieuse école de commerce. ***Service lecteur n°12.***

■ Promotion sur le système portable Dell 316LT, équipé d'un 386sx. 28 950 FHT pour l'unité centrale, avec un disque dur de 40˘Mo, une souris, le logiciel Works˘2.0, et... une housse. ***Service lecteur n°13.***

■ Toshiba propose une version avec disque dur de 80 Mo de son portable T3100SX, au prix de 46 990 FHT. ***Service lecteur n°14.***

■ Le Mitac LCE 386SX-20 utilise un 386sx cadencé à 20 MHz. Disponible en deux versions, avec disque dur de 40 Mo (24 700 FH.) ou 80 Mo (27 600 FHT), le LCE 386SX-20 dispose de 1 Mo de RAM, et un contrôleur Super VGA. ***Service lecteur n°15.***

50 GRAMMES D'INTELLIGENCE, ÇA N'A PAS DE PRÍX.

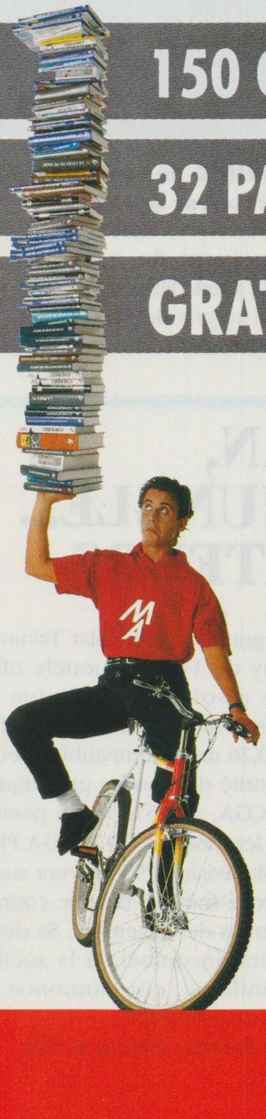


LE NOUVEAU CATALOGUE
MICRO APPLICATION EST ARRIVÉ.

150 OUVRAGES

32 PAGES COULEUR

GRATUIT



WELLDONE

Toutes les informations sur les sujets clés de l'actualité.

Disque dur • Imprimantes • Novell • Entretien et réparation • PC 386 • Windows 3 • MS-DOS 3.3 et 4.01 • Norton Utilities 4.5 • PC Tools deluxe 6 • Programmation Systeme • Basic • Turbo Pascal • Turbo C ++ • Turbo Assembleur • QuickBasic • Cartes VGA • Word 5 • Word pour Windows • Corel Draw • Harvard Graphics • PC Paintbrush • Lotus • Superbase • Multiplan 4 • dBase • Excel • MS/Works 2 • Virus • Flight Simulator • Larry • Deluxe Paint • Stos Basic • GFA Basic • Domaine Public ...

N'hésitez pas à le demander à votre revendeur ou à nous contacter sur 36-15 MICROAPP ou 58 rue du Faubourg Poissonnière 75010 Paris.

Cerclez info pub 07

COMME C'EST BON D'ETRE INTELLIGENT

FLASH... OH, OH !

Intel a annoncé la première mémoire Flash de 2 mégabits. Rappelons que la mémoire Flash occupe une place particulière entre les deux grandes familles de mémoires : vives (RAM) et mortes (ROM). Les mémoires RAM (Random Access Memory) ne conservent pas d'informations si l'alimentation électrique est supprimée. Il est possible d'accéder à ces mémoires tant en lecture qu'en écriture. Deux sortes de RAM sont actuellement utilisées. La DRAM ou RAM dynamique ne conserve ses données que pendant un temps assez court, des circuits spéciaux de rafraîchissement sont indispensables pour éviter la perte des données. Cette mémoire vive est devenue relativement bon marché et constitue la mémoire principale de tous les PC actuels. La deuxième sorte de mémoire vive est la SRAM, ou RAM statique. Elle conserve ses données jusqu'à la suppression de l'alimentation. L'avantage principal de cette sorte de mémoire est son temps d'accès très réduit. Malheureusement, elle reste très chère. C'est la mémoire vive qui est utilisée dans les caches mémoire.

D'un autre côté, les mémoires ROM (Read Only Memory) conservent les informations en permanence, même privées d'alimentation électrique, mais il n'est pas possible à l'utilisateur d'y inscrire d'informations. Ces dernières sont gravées lors de la fabrication de la mémoire. Les mémoires PROM sont des mémoires ROM qui permettent une seule et unique écriture. Enfin, les mémoires EPROM ressemblent aux PROM, mais il est possible d'écrire plusieurs fois des données. Les premières EPROM, pour être reprogrammées, devaient d'abord subir une insolation d'ultraviolets pendant 2 à 3 minutes. Les mémoires Flash appartiennent à cette dernière catégorie de mémoire, le bain d'UV en moins. L'écriture se réalise en 150 nanosecondes. Les avantages de la mémoires Flash sont donc multiples : conservation des données si l'alimentation électrique est supprimée, écriture multiple possible, accès instantané à l'information mémorisée (d'où son nom). Les applications sont nombreuses. En premier, cette mémoire pourra remplacer les mémoires vives statiques alimentées par batterie. Ensuite, l'utilisation dans les portables est envisageable pour remplacer les disques (économie importante de courant, il faut presque trois watts pour procéder à une lecture sur un disque dur de 3,5 pouces, alors qu'une mémoire flash ne réclame que soixante milliwatts). Les autres applications possibles sont principalement dues au nouveau boîtier proposé par Intel.

Ce nouveau boîtier extramince (1,2 mm d'épaisseur) a pour nom TSOP (Thin, Small Outline Package). Par comparaison, il n'occupe que le tiers ou la moitié du volume des boîtiers pour montage en surface. Les applications pouvant profiter d'un tel boîtier sont nombreux : les cartes mémoires, les instruments portables, les ordinateurs portables, les unités de disques et les équipements de communication portatifs. Les mémoires Flash de 1 Mbits et de 2 Mbits seront disponibles dans ce nouveau boîtier.

La troisième annonce réalisée par Intel concerne le 80186. Une nouvelle version dénommée 80C186EB sera bientôt disponible pour réaliser des circuits ASSP (Application Spécific Standard Product) ou ASIC (circuits intégrés sur mesure) grâce à une bibliothèque de cellules en un micron (la largeur minimale des fils de poly ou d'aluminium sont d'un micron, la précédente largeur était de 1,2 micron — la nouvelle mémoire Flash bénéficie de cette nouvelle technologie à un micron). La caractéristique principale de ce nouveau 80C186EB est de posséder un cœur entièrement statique. On peut couper l'alimentation sans perdre le fil du programme en cours. Dès que le circuit est réalimenté, le processeur reprend son travail là où il l'avait laissé, comme si la coupure d'électricité n'avait pas eu lieu. Les domaines d'application sont aussi très variés (contraintes de consommation et d'encombrement) : téléphones cellulaires numériques, télécopieurs portables, lecteurs de codes barres portables, organiseurs électroniques, etc. Une version à bus externe de 8 bits et bus interne de 16 bits, le 80C188EB, est aussi disponible. Une version militaire du 80C186EB sera prochainement proposée. Intel fournit, en plus de la bibliothèque de cellules, différents outils pour la simulation et la mise au point d'applications intégrant le nouveau processeur.

Service lecteur n° 16

14 POUCES POUR CONVAINCRE

Le dernier-né de la gamme des moniteurs Hitachi, le modèle 14MVX, est un écran couleur de 14", multisync (ou multinorme). Il est en effet compatible avec les modes graphiques MCGA, VGA, SuperVGA, 8514/A (résolution 1024 x 768 points).

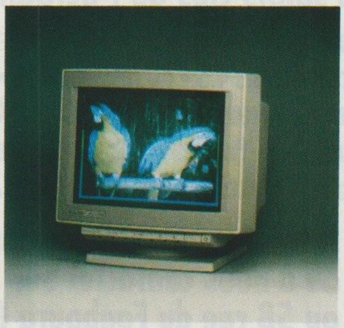


Le 14MVX, un moniteur multisync couleur de 14".

Le 14MVX dispose d'un écran à coins carrés, traité antireflet. Le pied est orientable et amovible, et l'ensemble des commandes (luminosité, contraste, taille et position verticale et horizontale de l'image) est regroupé sur la face avant du moniteur. Le 14MVX est dès à présent disponible au prix de 5 990 FHT.

Service lecteur n° 17

TAXAN, ROI DE LA JUNGLE... DES MONITEURS



Un écran couleur multisync de 14".

Avec le Multivision 795, Taxan propose un moniteur multisync intégrant un tube plat Trinitron Sony de 14". Ce modèle offre une résolution maximum de 1024 x 768 points et un « pitch » de 0,26 mm. Compatible avec la majorité des modes graphiques, du CGA, au 8514/A, en passant par les modes VGA et VGA Plus, le Multivision 795 arrivera sur le marché français dans le courant du mois de septembre. Sa distribution incombera à la société Omnilogic, qui l'annonce au prix de 6950 FHT.

Service lecteur n° 18

CANON JOUE LA CARTE TELFONT

La carte Telfont, développée par la société Interval, améliore les performances des imprimantes à laser Canon série III. Son action consiste à stocker les éléments répétitifs (logos, fonds de pages, en-têtes de lettres, etc.) sous formes de fichiers compressés. Intégrée à l'imprimante, la carte Telfont est gérée comme un disque dur, et une simple commande provoque l'envoi des données vers l'imprimante. Telfont contient également des programmes, dont un de génération de codes à barres. Selon le type du fichier, la compression des données permet des gains de l'ordre de 30 à 90 %. La carte Telfont existe en version équipée d'une mémoire de 1 Mo (6 500 FHT) ou 512 Ko (5 000 FHT).

Service lecteur n° 32

UNE NOUVELLE QMS POSTSCRIPT

La QMS PS 410 représente la nouvelle entrée dans la gamme des imprimantes à laser PostScript. Moyennant 24 900 FHT, l'utilisateur de la QMS PS 410 bénéficie d'une imprimante PostScript, dotée, de surcroît, d'une émulation résidante HP LaserJet série II (HPPCL niveau 4). Une carte additionnelle permet d'ajouter les possibilités du langage HPGL à la panoplie de la PS 410. L'imprimante émule le mode adéquat de manière automatique, en fonction du fichier à traiter.

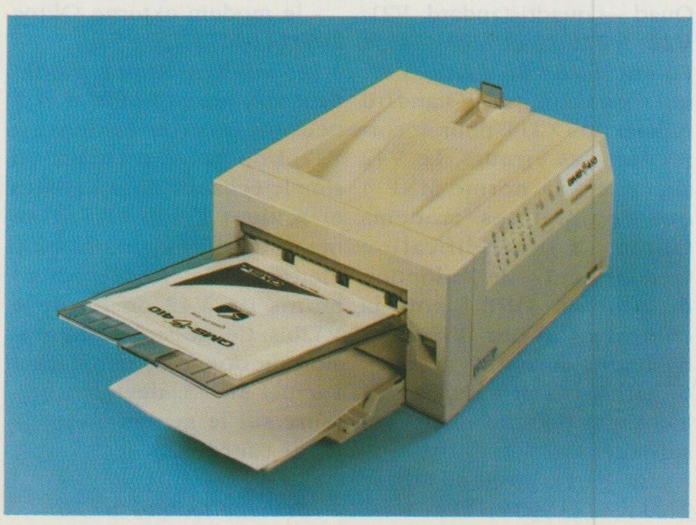
Ce modèle compte trois connecteurs : une interface série RS-232, une prise parallèle, et un port Apple Talk.

Disposant de 45 polices de caractères Adobe PostScript résidantes (outre les « 35 » standard s'ajoutent l'Helvetica Narrow et le Garamond), la QMS PS 410 peut s'enrichir de polices supplémentaires (téléchargées ou cartouches). L'émulation HP LaserJet lui donne accès à l'ensemble de la typothèque de caractères HP série II.

Techniquement, la QMS PS 410 comporte une base mécanique Canon LX et un Motorola 68020 à 16,7 MHz. Doté de 2 Mo de mémoire vive (extensible à 6 Mo), la dernière-née de la gamme QMS imprime quatre pages par minute.

Service lecteur n° 33

L'imprimante QMS PS 410.





Photos non contractuelles — Publicité réalisée en PAO par PAONGRAPHI® 42 77 80 80

Entrez dans le réseau de distribution aAT.

Une très large gamme d'ordinateurs et de périphériques vous est offerte, du PC de base avec un processeur 8088, au très puissant 486.

Contactez-nous dès maintenant au :

48 43 75 63

aAT

Intelligence Computer Europe
99/101, rue Henri Gautier
93000 BOBIGNY

☎ 48 43 75 63
Fax 48 43 75 74

Cerclez info pub 08

TECMAR TIRE SA DERNIERE CARTOUCHE



Jusqu'à 1 Go sur des cartouches DC 600.

Et quelle cartouche ! L'unité de sauvegarde sur bande QT-1050es, qui utilise des cartouches de type DC 600, revendique une capacité de stockage supérieure à 1 Go, exactement 1,05 Go. Le QT-1050es est géré par le logiciel QTOS, compatible avec le DOS, OS/2, et les réseaux Novell. Cette unité est dotée d'une interface SCSI, et dispose d'un système de correction d'erreur ECC. Bien entendu, elle peut relire toutes les cartouches au format DC 600. Le logiciel QTOS permet, par compression des données, de doubler les capacités du QT-1050es : 2,1 Go. Sa distribution est assurée par la société Omnilogic, qui le propose au prix de 45 000 FHT.

Service lecteur n° 22

JUSQU'A 200 MO DE MEMOIRE EXTERNE

Les disques durs externes Datafile existent en trois versions, d'une capacité de 40 Mo formaté à 120 Mo. Ils se connectent directement sur l'interface parallèle des PC. Ils sont livrés avec un logiciel permettant une partition aisée. Ils pèsent moins de 1 kg, et disposent d'une batterie Nicad lui assurant deux heures d'autonomie. Fabriqués par la société Axonix, les Datafile sont distribués par Potentiel Diffusion. Au catalogue figurent les modèles de 40 Mo (8 500 FHT), de 100 Mo (13 500 FHT), et de 200 Mo (21 000 FHT).

Service lecteur n° 23

COMMUNICATION A LA CARTE

Le modem Quintet de la société belge Telindus est basé sur le concept de modularité, original pour ce type de produit. A partir d'une carte mère, le Quintet évolue en fonction des modules encastrables qui lui sont adjoints. Ainsi, le Quintet peut tour à tour devenir un modem multimode V32, V22 bis, V22, Bell 212A, Bell 103, une carte X25, une carte RNIS, ou encore un coprocesseur de communication intégrant un modem X25 et 3270. Il peut également recevoir de la mémoire supplémentaire et un module d'encryptage.

Le modem Quintet, disponible en version carte ou sous forme de boîtier externe, dispose d'un logiciel de communication vidéotex européen compatible avec les normes Prestel (Grande-Bretagne), Télétel, TTY, et Bildschirmtext (utilisée en RFA), et du logiciel de communication Tempo.

Les tarifs du modem Quintet sont de 11 400 FHT pour la version carte, et 11 800 FHT pour le modèle externe.

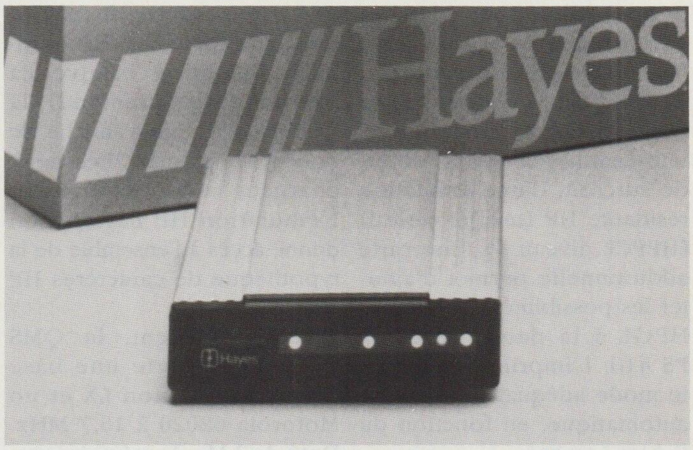
Service lecteur n° 24

UN MODEM TRES « SMART »

Le Smartmodem 1200 arrive sur le marché français, et reçoit l'agrément des PTT. Il s'agit d'un modèle tristandard V21, V22, et V23, fonctionnant en mode synchrone ou asynchrone. Il est compatible avec la norme Hayes, et reçoit donc le jeu de commandes correspondantes. Métrologie propose le Smartmodem 1200 sous deux formes, carte PC (Smartmodem 1200B, 4 300 FHT), et boîtier externe (Smartmodem 1200, 4 800 FHT).

Service lecteur n° 25

Le Smartmodem 1200, modem tristandard compatible Hayes.



OLITEC ET LE V42

A compter du mois de septembre, les modems Olitec Quad — quadristandard, V21, V22, V22bis, et V23) — se voient dotés du mode V42, qui n'est autre que le standard officiel du CCITT pour la correction des erreurs. Le V42 succède à la norme MNP 4 qui équipait les modems Olitec. Cette modification (accompagnée de l'adoption de la norme MNP 5 pour la compression des données) n'entraîne aucun changement de tarifs. Les prix demeurent inchangés pour les références suivantes :

- la carte modem Olitec 2400 PS/2 (3 990 FHT) ;
- la carte Olitec 2400 PC (3.690 FHT) ;
- le modem externe Olitec 2400 (4 690 FHT).

Les modems Olitec répondant aux normes MNP 4 et 5 peuvent recevoir l'extension V42. L'opération nécessite un simple échange de plaquettes et revient à 600 FHT. Une manipulation similaire permettra de doter les modems munis du V42 de l'option V42bis. Ce mode de compression de données, remplaçant l'actuel MNP5, devrait apparaître sur le marché avant la fin de l'année.

Service lecteur n° 26

"Killer Price And A Terrific Warranty"

PC Magazine/July 1990

Polywell Computers

Constructeur Américain

de notoriété Internationale,
vous propose :
10 Configurations de Basz (TOP 10) #
du 286 au 486-33 Mhz

- # Garantie 5 ANS
- # Livraison sous 5 JOURS
- # Programme Distributeur

(nous consulter)

PC MAGAZINE EDITOR'S CHOICE

INFO WORLD TOP 100 1989



POLY 386-25 Mhz/Cache
4 Mo de RAM
Disque Dur 105 Mo
Carte V.G.A. "Orchid"
(1.024 × 768) 1 Mo
Ecran Multisync "SONY" "14"
Prix : 30.892 F H.T.

POLY 386-33 Mhz/Cache
+ 1.700 F H.T.

POLY 486-33 Mhz/Cache
+ 31.385 F H.T.

POLY 386-sx 20 Mhz
à partir de : 9.077 F H.T.

POLY 286-16 Mhz
à partir de : 6.522 F H.T.

Pour une Configuration
"SUR MESURE"
Consultez-nous (appel gratuit)

NOS GARANTIES :
- 5 ANS de Main d'Oeuvre gratuite retour atelier
- 2 ANS Cartes Mères
- 1 AN Pièces & Main d'Oeuvre
- 30 JOURS satisfait ou remboursé
- 1 AN de Maintenance sur site GRATUITE

Pour de plus amples informations,
Veuillez nous contacter au :
Numéro Vert
05.11.66.88
Appel GRATUIT
sur toute la FRANCE

POLY SYSTEMS

Polywell Europe
Central Parc Villepinte
255, Bld Robert Ballanger
2, Allée du Cerf
93420 – VILLEPINTE
RER (B3)
Autoroute : A1 ou A3
HOT LINE : (1) 49.63.32.88
FAX : (1) 48.61.18.26

ouvert du Lundi au Samedi
de 9 H 30 à 19 H 00

TOSHIBA EN BAISSE

Cette chute ne concerne que les seuls tarifs des modems de la marque, destinés à équiper les portables du constructeur japonais. L'ensemble de la gamme subit ce réajustement. En voici les exemples les plus significatifs :

- TM Arkansas, 1 790 FHT (au lieu de 1 850 F) ;

- TM EA 2400, 4 490 FHT (au lieu de 5 950 F) ;

- TMTRC 3270, 9 990 FHT (contre 11 950 F).

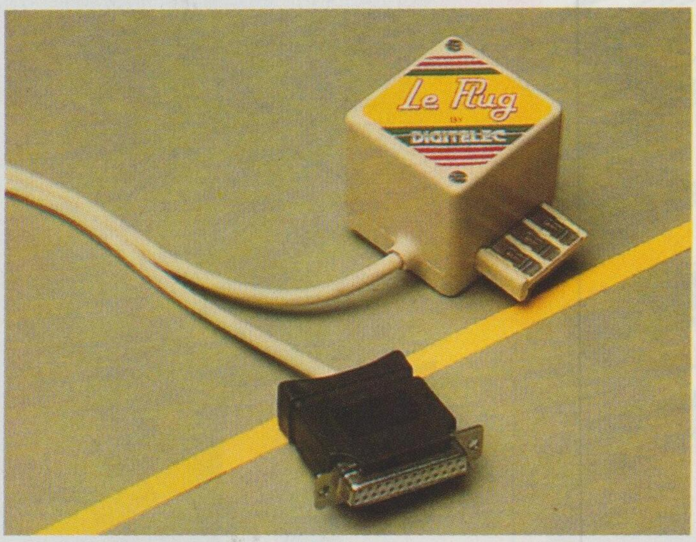
Service lecteur n° 27

IL ETAIT UN PETIT MODEM

Ce petit boîtier (dimensions : 52 x 52 x 49 mm), muni d'une prise de téléphone gigogne, renferme un véritable modem. Présenté par ses concepteurs, la société Digitelec, comme le plus petit modem du monde, le Plug est un modem tristandard, V21, V22, et V23. Le Plug se connecte directement à l'interface RS-232 de l'unité centrale, et son alimentation électrique est assurée par la prise de téléphone. Le Plug est compatible avec la norme Hayes. Il gère l'appel et les réponses automatiques, et intègre une option de mise en veille automatique. Le prix public du modem Plug est fixé à 2 490 FHT.

Service lecteur n° 28

L'alimentation électrique du Plug est assurée par la prise de téléphone.



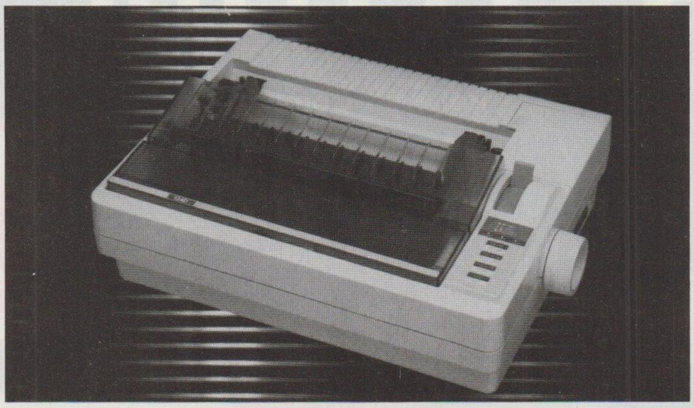
LE PIQUANT DE LA DL-1100

Fujitsu propose une nouvelle imprimante matricielle à 24 aiguilles et 110 colonnes. La DL-1100, tel est son nom, offre une vitesse d'impression de 60 cps en qualité courrier, ou 240 cps en qualité courante. L'impression de codes à barres ne nécessite aucune extension. La DL-1100 comprend huit polices de caractères résidantes, et trois modes d'émulation : IBM Proprinter XL24, Epson LQ 2500, et Fujitsu DPL 24C Plus.

Dotée d'une mémoire tampon de 24 Ko, l'imprimante DL-1100 est proposée au prix de 4 490 FHT. Fujitsu commercialise diverses options, dont un kit couleur, un bac d'introduction feuille à feuille, et de nombreuses cartes de polices additionnelles.

Service lecteur n° 29

CITIZEN 124D : 24 AIGUILLES POUR 3 250 FHT



Une imprimante matricielle à 24 aiguilles au meilleur prix.

Citizen commercialise depuis le début du mois de septembre une imprimante à 24 aiguilles d'entrée de gamme. La 124D est un modèle 80 colonnes, doté des émulations Epson LQ, IBM Proprinter, et NEC graphique. La vitesse d'impression est de 120 cps en mode listing et 40 cps en qualité courrier. La mémoire tampon standard de 8 Ko peut être portée à 32 Ko (prix de l'extension : 127 FHT). La 124D est susceptible de recevoir un introducteur feuille à feuille automatique (990 FHT), ou semi-automatique (490 F), ainsi qu'une interface série RS-232 (360 F).

Service lecteur n° 30

IMPRIMANTE POIDS PLUME

Les atouts de la Litewrite résident dans son faible encombrement, sa légèreté (moins de 2 kg), et dans son autonomie de 85 pages, assurée par une batterie amovible. La Litewrite est une imprimante matricielle à neuf aiguilles, disposant des émulations Epson et IBM, et munie d'une interface série et d'un port parallèle. La vitesse d'impression atteint 100 cps en qualité courante, contre 25 cps en qualité courrier. Imprimante 80 colonnes, la Litewrite devient une 136 colonnes en mode condensé. La Litewrite est disponible auprès de la société Potentiel Diffusion, au prix de 4 560 FHT.

Service lecteur n° 31

Litewrite : légère et autonome.



CARTES GRAPHIQUES

Au début de l'été, le fabricant de cartes Rasterex dévoilait une gamme inédite de cartes graphiques accélératrices haute définition. Dénommée Liberty, la nouvelle famille abrite des cartes pour PC ou PS/2, offrant des définitions de 1024 x 768 points jusqu'à 1280 x 1024 points. Les différents modèles disposent de 1 à 16 Mo de mémoire. Elles sont également compatibles avec le mode VGA. ISTC se charge de la distribution des cartes Liberty, dont les prix s'échelonnent de 25 500 FHT, à 40 600 FHT.

Service lecteur n° 19

JUSQU'A 120 MO DANS LA POCHE

La série « M » des micro-ordinateurs Victor reçoit l'apport des Add Pak 3. Il s'agit de la dernière génération des disques durs amovibles, qui se positionnent sur la face avant des PC série « M », équipés en série d'un connecteur dédié. L'unité centrale détecte automatiquement la présence et la capacité des Add Pak 3. Ces disques durs sont disponibles en deux versions, de 40 Mo formatés (4 990 FHT), et de 80 Mo formatés (7 990 FHT). Un modèle de 120 Mo complétera la gamme dans le courant du mois d'octobre. Son prix se situera autour de 11 000 FHT.

Service lecteur n° 20

LA SAUVEGARDE DONNE DE LA BANDE

Almatec assure la distribution des unités de sauvegarde sur bande de la série MiniVault 120, conçue par Tecmar. Destinées aux PC, XT et AT, ainsi qu'aux PS/2, les MiniVault utilisent des cartouches au format d'un quart de pouce de 80 ou 120 Mo formatés. Les possibilités de stockage peuvent être doublées grâce au logiciel de compression de données Tecmar QTOS, livré en standard avec les unités de la famille MiniVault. Famille constituée de trois modèles, l'un externe, le MiniVault 120e, vendu 7 950 FHT, les deux autres internes, les 120i pour PC, et 120i/PS pour PS/2, tous deux disponibles au prix de 6 290 FHT.

Service lecteur n° 21

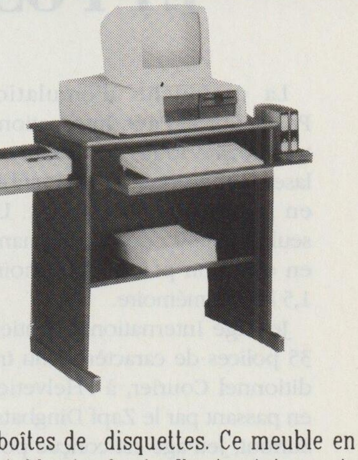
Le MiniVault en version externe et interne.



ALCO SYSTEME
L'ACCESSOIRE INFORMATIQUE

789 F TTC

L'INTEGRAL MICRO



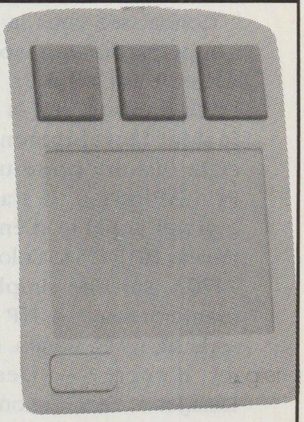
Poste de travail informatique adapté à tout équipement micro. Deux tablettes latérales coulissantes (41 x 29 et 25 x 29 cm) permettent de disposer aisément une imprimante et des boîtes de disquettes. Ce meuble en bois, de qualité professionnel, est stable et robuste. Il est peint en gris antistatique et anti-reflet. Dimensions: H 79 cm. Plateau: L 77 x 41,5 cm. Livré prêt à monter.

MEUBLE COMPLEMENTAIRE POUR IMPRIMANTE

Monté sur roulettes, stable et robuste, il bénéfice d'une peinture grise traitée antireflet et antistatique. Deux étagères pour la réception et le stockage des listings. Fente pour le passage du papier. Dimensions: H80 x L60 x P40 cm. Livré prêt à monter. **689 F** TTC

669 F TTC

SOURIS TACTILE AGILER



Première souris à écran tactile. Vous déplacez le curseur par un simple mouvement du doigt sur l'écran tactile de 52 x 42 mm. Plus besoin de bouger la souris. Idéal sur un espace de travail réduit ou avec un portable. Robuste et fiable. Livré avec des programmes de démonstration. Vitesse de déplacement du curseur jusqu'à 500 mm/s. Résolution: 200 à 1000 DPI, dynamique. Pour PC et compatibles.

SOURIS INFRA ROUGE SANS FILS.

Cette souris, pour PC et compatibles, vous donnera une liberté d'utilisation que vous ne trouverez jamais avec aucune autre souris traditionnelle. Compatible microsoft. **669 F** TTC

PROMOTION SUR LES DISQUETTES GOLDSTAR

Boite de 10 disquettes 5"1/4 2F2D 48 TPI certifiées sans erreurs...69 F
Boite de 10 disquettes 3"1/2 2F2D 135 TPI certifiées sans erreurs.99 F

A retourner avec votre règlement à: ALCO SYSTEME
36 Avenue d' Italie 75013 Paris Tél: 45 88 90 30 Fax: 45 88 90 40

Nom..Prénom..........................
Adresse..
Ville...Code postal......................

Désignation	Prix	Qté	Total
Participation frais de Port			
TOTAL TTC			

☐ Je désire un catalogue
☐ Je passe une commande

Mode de règlement:
☐ Chèque ou CCP
☐ Mandat
☐ Contre remboursement +40F

Port 39F sans mobilier, 99F avec un meuble, 69F par meuble supp.

Cerclez info pub 10

LASERJET ET POSTSCRIPT

La cartouche d'émulation PostScript JetPage International transforme toute imprimante à laser HP LaserJet, ou compatible, en imprimante PostScript. Un seul impératif : que l'imprimante en question possède au moins 1,5 Mo de mémoire.

JetPage International contient 35 polices de caractères (du traditionnel Courier, à l'Helvetica, en passant par le Zapf Dingbats). Surtout, JetPage est conçue pour le marché européen, et possède à ce titre l'ensemble des caractères accentués en usage dans ces pays, ainsi que les jeux de caractères internationaux.

JetPage fonctionne avec l'ensemble des applications du marché, dont PageMaker, Ventura, Corel Draw !, Word 5, Illustrator, WordPerfect, Windows, AutoCAD, etc. La carte JetPage International est destinée aux imprimantes HP LaserJet, séries II, IIP/IID, et III, ainsi qu'aux Canon LBP-8 II, Olivetti PG 108, et Siemens PT 10, 7500 et 7800.

JetPage International est édité par Computer Peripherals.

Service lecteur n° 34

ET L'IMPRIMANTE DEVINT TRACEUR

Connaissez-vous la fable de l'imprimante qui souhaitait devenir un traceur ? Due à l'imagination de la société Insight Development Corp., cette histoire porte un nom : PrintAPlot 2.0. Il s'agit d'un logiciel capable d'émuler un traceur HP 7475A, Color Pro, ou 7470A, sur une simple imprimante à laser — HP LaserJet série III, QMS, Canon, etc. —, à jet d'encre — DeskJet et DeskJet +, Dec, Canon, etc. —, ou matricielle à aiguilles. Particulièrement destinée aux utilisateurs de logiciels de CAO/DAO, AutoCAD par exemple, PrintAPlot ne nécessite que 256 Ko de mémoire vive sur l'unité centrale.

La simulation du traceur est poussée, le logiciel permettant de paramétrer le positionnement du tracé sur la feuille, la taille de la plume, son modèle et sa couleur. PrintAPlot gère l'agrandissement (jusqu'à 600 %) et la réduction (jusqu'à 1 %), et les effets de rotation des tracés.

La distribution de ce logiciel est assurée par la société EDN. PrintAPlot 2.0 coûte 3 600 FHT en version francisée, alors que la mise à niveau destinée à la version 1.0 est proposée au prix de 2 000 FHT.

Service lecteur n° 35

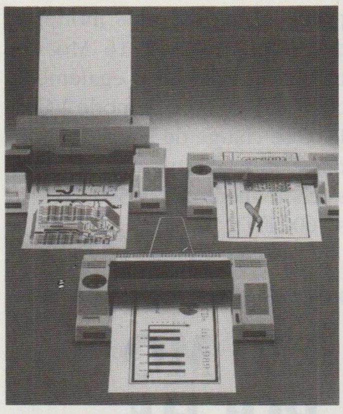
SUIVEZ LA TRACE

Pluie de nouveautés chez Calcomp qui dévoile ses nouveaux modèles de traceurs. Toutes les familles de la gamme sont touchées par cette opération de renouvellement. A commencer par la série des traceurs électrostatiques monochromes 57000, qui accueillent trois modèles : les 57424, au format A1 (218 000 FHT), 57436, au format A0 (256 000 FHT), et 57444, format 44" supérieur au A0 (527 000 FHT). Le modèle Drawingmaster 52236 adopte quant à lui la technologie du transfert direct, garante d'une grande fiabilité et de vitesses élevées. Ce traceur au format A0 est équipé de trois interfaces série RS-232, et d'un port parallèle Centronics. Il est commercialisé depuis le mois de juillet au prix de 189 500 FHT.

Plus modeste, le modèle 1026, toujours au format A0, est un traceur à plumes, alimentation en feuille à feuille ou en rouleau, vendu 77 100 FHT.

Service lecteur n° 36

LA PLUME SINON RIEN

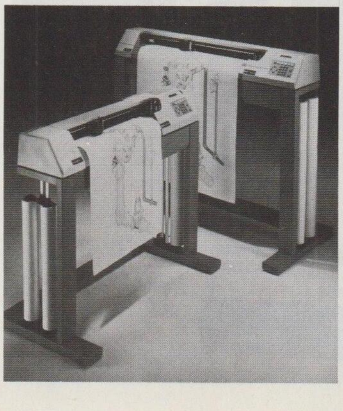


La série G1000 : des traceurs « compacts » à partir de 8 000 FHT.

Neuf nouveaux modèles viennent enrichir la gamme des traceurs à plumes d'Océ Graphics.

Il s'agit tout d'abord de deux traceurs petit format (A3/A4) destinés en priorité à la présentation de statistiques, la conception de schémas filaires, de tracés d'essais, de graphiques de gestion. L'atout de ce type d'appareils réside essentiellement dans leurs coûts et un encombrement réduit. Ils n'occupent, en effet, guère plus de place qu'une imprimante (120 x 495 x 280 mm) et pèsent 7 kg. Les deux modèles dévoilés, référencés G1012 et G1052, ont de nombreuses caractéristiques en commun. Ils utilisent indifféremment du papier, du film transparent, ou encore du calque. Ils peuvent être équipés de pointes en céramique, de feutres, ou de plumes techniques jetables de type HP. La résolution atteint 0,025 mm, et le niveau de bruit ne dépasse pas 58 dB. Les traceurs de la série G possèdent une interface série RS-232 et sont compatibles HGPL. Quatre éléments permettent de différencier les modèles G1012 et G1052 : le nombre de plumes (6 pour le G1012 contre 8 pour le G1052), la vitesse de tracé (42 cm/s contre 92 cm/s), la capacité de la mémoire tampon (respectivement 1 Ko et 1 Mo), et enfin le prix. Le G1012 coûte en effet 8 100 FHT, alors que le G1052 est annoncé à 16 500 FHT.

G1830-C, tracés de plans au format A0 ou A1.



Simultanément, Océ Graphics a dévoilé une nouvelle famille de traceurs grands formats A0 ou A1, la série G1800-C, forte de sept modèles Leur domaine d'expression privilégié demeure l'exécution de plans. Cette gamme, équipée de huit plumes, est dotée de deux interfaces RS-232 en standard. Elle utilise tous types de support papier, ordinaire, couché, translucide, calque, ainsi que les films transparents, polypropylène, polyester et triplot. La résolution maximum atteint 0,125 mm.

Le premier modèle, G1824-C, est au format A1, il fait appel à une alimentation feuille à feuille. Il trace à la vitesse de 120 cm/s, et jusqu'à 1,3 m. Son prix est de 46 000 FHT. Il est doté de 100 Ko de mémoire tampon. Suit le G1825-C, identique à l'exception du format (A0) et du prix (59 500 FHT). Océ Graphics propose ensuite les G1834-C et G1834-CR, au format A1. Ils possèdent 4 Mo de mémoire tampon et atteignent une vitesse de 140 cm/s. Le G1834-C (56 100 FHT) est alimenté exclusivement en feuille à feuille, alors que le G1834-CR peut également utiliser des rouleaux de papier.

Les G1835-C (78 000 F) et G1835-CR 89 000 F) sont les versions au format A0 des deux précédents modèles.

Le sommet de la gamme est occupé par le G1845-AC (99 000 FHT), traceur au format A0, alimenté en feuille à feuille ou en rouleau. Il se différencie par des options de découpe et de rangement automatique des plans, et de reconnaissance des plumes.

Service lecteur n° 37

MOTOROLA DEVIENT MEMBRE PRINCIPAL D'UNIX INTERNATIONAL

A ce titre Motorola participera aux comités exécutifs et d'organisation d'Unix International. Rappelons que cette association, qui veille sur la destinée d'Unix, a été créée en décembre 1988 et regroupe à ce jour 155 membres (représentant plus de 80 % de la base installée d'Unix System V), dont quatorze principaux. Les trois autres principales sociétés de développement de microprocesseurs (Intel Corporation, MIPS et Sun Microsystems) sont également membres de l'association et contribuent à la percée d'Unix System V.

Service lecteur n° 38

LA FEE CUMULUS TRANSFORME LES 286 EN 386SX

La MAXICARD 386SX de Cumulus fait évoluer les micro-ordinateurs à base de 80286, en remplaçant leur processeur par un i386sx à 16 MHz. Elle permet d'utiliser tous les produits spécifiques au i386. L'installation de la Maxicard est simple et ne requiert aucun connecteur libre : il suffit de l'insérer en remplacement du processeur 80286. Un emplacement pour le 387sx est disponible sur la carte. Pour 5 000 FHT, (plus le kit d'installation, notamment l'extracteur de « chip » pour enlever l'ancien 286) n'importe quel AT essoufflé peut désormais se transformer en un 386sx à 16 MHz flambant neuf.

Service lecteur n° 39

TIGER : UNE CARTE MEMOIRE UNIVERSELLE

A quoi servent les cartes mémoires ? Soit à étendre la mémoire vive de votre PC pour profiter des bienfaits de la mémoire EMS, des caches disques, disques virtuels, etc. On peut également interposer une carte mémoire entre les entrées/sorties et un périphérique. Les données s'accumulent dans la mémoire de la carte au lieu de monopoliser l'unité centrale. Ou encore, certaines applications peuvent nécessiter un « buffer image » pour le traitement de fichiers graphiques volumineux. La carte TIGer de ACE, destinée aux micro-ordinateurs à bus ISA, permet d'exploiter les cartes mémoire de chacune de ces manières.

Compatible avec les scanners AVR-302, AVR 3000/GS, Canon IX-12, etc., ainsi qu'avec toute imprimante pourvue d'une interface vidéo, elle permet de numériser un document A4 en 300 dpi en neuf secondes, et dix secondes suffisent à une sortie sur HP LaserJet série II. Son débit nominal atteint 1,6 Mbits par seconde.

TIGer existe en deux versions : TIGerCUB qui ne possède pas de mémoire propre et se contente d'exploiter la mémoire EMS déjà présente sur votre système, et TIGerEMS possédant 2 Mo de base, extensible à 8 Mo.

Servicer lecteur n° 40

T'AS DE BEAUX YEUX, TU SAIS !

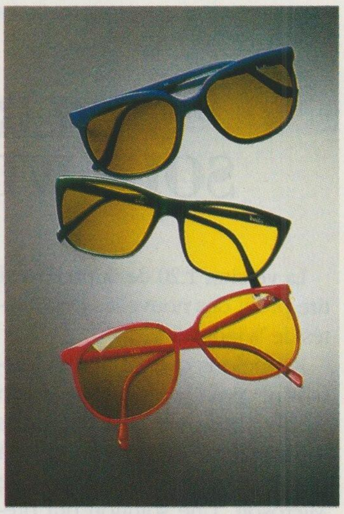
Comment lutter efficacement contre les agressions des sources de lumières artificielles, des écrans de tous poils (télévisions, moniteurs informatiques, etc.) sinon en adoptant la protection d'une paire de lunettes adaptée ?

C'est du moins la philosophie de la société Bollé, célèbre lunetier, qui propose une large gamme de modèles basée sur la technologie ASV (Anti Stress Vision). Le principe consiste à obtenir des verres capables d'éliminer les reflets, de filtrer les rayons ultraviolets, principalement ceux qui émanent des néons, et de supprimer jusqu'à 90 % des rayons infrarouges résiduels.

En plus d'être utiles, les lunettes ASV de Bollé, sont élégantes. Disponibles en plusieurs couleurs, elles se déclinent en cinq modèles différents, classiques ou plus audacieuses. Leur prix se situe autour de 280 FTTC.

Service lecteur n° 41

Lunettes Bollé : vos yeux ont de la chance.



SPOOLER MULTISERVEUR

Le spooler Bytelink + complète l'offre de la société Numetronic en matière de spooler graphiques multiserveur.

Le Bytelink + dispose de 24 ports, en série ou en parallèle. Les ports séries sont bidirectionnels et permettent de gérer les tables traçantes aux formats A4 à A0. Un microprocesseur intégré assure le transfert des données à une vitesse de 21 000 bauds. Comme les autres modèles Bytelink (4, 8, 12 ports), le spooler Bytelink + est entièrement compatible avec les logiciels courants, Word 5, dBase, ou Lotus 1-2-3. Le prix de ce modèle est de 30 950 FHT.

Service lecteur n° 42

■ Les disques durs amovibles Disk Pack d'IEF, de 42 Mo jusqu'à 200 Mo de capacité, sont distribués par la société Unisys. ***Service lecteur n° 43***

■ La carte Brother CB8 PS transforme les imprimantes à laser Brother HL-8 et HL-8e en imprimantes PostScript. En plus du langage de description de page, ce kit comprend 2 Mo de mémoire supplémentaire. La carte CB8 PS est vendue 14 500 FHT. ***Service lecteur n° 44***

■ L'imprimante à laser Nec SW 266 pour 18 450 FHT, depuis le début de l'été. Ce modèle à émulation HP LaserJet II, offre une vitesse d'impression de 8 pages par minute. ***Service lecteur n° 45***

■ Le moniteur Moniterm Viking 2LC est un écran monochrome de 19". Il dispose d'une large bibliothèque de drivers (Windows, Word 5, Ventura, Lotus 1-2-3, etc.). Il est compatible avec les modes Hercules et CGA. Distribué par MTE, le Viking 2LC coûte 14 950 FHT. ***Service lecteur n° 46***

■ Ace propose un écran pleine page de 19" AceView couleur. Compatible avec les modes graphiques CGA, EGA, VGA, Super VGA, et 8514/A, l'AceView est livré avec une carte Super VGA 16 bits de 512 Ko de mémoire (14 980 FHT carte comprise). ***Service lecteur n° 47***

■ Ricoh présente un télécopieur du groupe IV à impression laser sur papier ordinaire. Le modèle 7000L/DG4 est vendu 135 000 FHT. ***Service lecteur n° 48***

■ Le télécopieur Sharp FO 5000 est un modèle à impression laser. Il appartient au groupe III et utilise du papier ordinaire. Vendu 82 783 FHT, le FO 5000 peut servir d'imprimante à laser. ***Service lecteur n° 49***

■ La carte d'interface TOPS FlashCard permet de raccorder tous les portables de la gamme Toshiba à des réseaux Macintosh. Elle est distribuée par La Commande Electronique (2 650 FHT). ***Service lecteur n° 50***

■ Mitsubishi propose un moniteur couleur multisync de... 25" ! il offre une résolution de 1280 x 1020 points, et un « pitch » de 0,31 mm. Yrel le commercialise au prix de 78 600 FHT. ***Service lecteur n° 51***

ACTUALITES LOGICIELS

SUPERBASE 4 SOUS WINDOWS 3

La version 1.20 de Superbase 4 tire profit des nouvelles possibilités de Windows 3.

Rappelons que ce SGBD, distribué par la société Micro Application, est l'un des rares logiciels de ce type adapté à l'environnement graphique de Microsoft. Une solution qui présente un certain nombre d'avantages — transparence de l'interface, intégration de fichiers graphiques, échanges dynamiques de données avec d'autres applications sous Windows.

En plus des traditionnels liens DDE (Dynamic Data Exchange) Superbase 4 tire avantage des connexions DLL (Dynamic Link Library). Ainsi, des librairies de fonctions additionnelles, créées indépendamment de Superbase, pourront être intégrées dans des applications de Superbase.

Les modifications les plus immédiatement perceptibles apportées à la version 1.20 de Superbase concernent l'aspect visuel — bouton en relief, etc. Egalement liées aux progrès de Windows, les nouvelles possibilités en matière d'adressage de la mémoire améliorent la vitesse de traitement des diverses opérations.

La compatibilité avec les formats graphiques BMP (Bitmap Windows 3) et EPS (Encapsulated PostScript) renforce l'avantage de Superbase dans le domaine de l'intégration d'images. Un besoin souvent exprimé par les professionnels de l'immobilier ou les vidéo-clubs, auquel ne répondent pas les SGBD classiques.

Enfin, Superbase 4 s'ouvre au SQL, via SQL-Connect et SQL-Vision, deux serveurs SQL développés par la société Henley Business Systems.

Superbase 4 est disponible depuis la mi-juillet en version française, au même prix que la version 1.0 (6 250 FHT).

Service lecteur n° 52

DB START, UN PRIX CANON

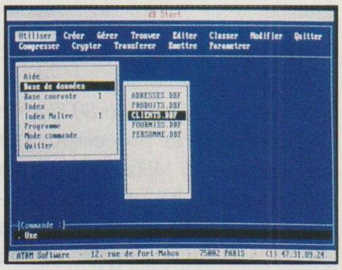
DB Start possède tous les attributs du parfait clone de dBase III Plus. Il est entièrement compatible, tant au niveau du mode Assist (l'interface utilisateur) que du langage de programmation, et proposé à un prix défiant toute concurrence : 450 FHT.

Une performance d'autant plus remarquable que dB Start se permet d'introduire des commandes inédites. Il s'agit en premier lieu de la commande Crypt, destinée à encoder les fichiers pour en garantir la protection. L'utilisateur dispose de trois modes de cryptage, avec clef ou sans clef, liés ou non à un poste de travail. La deuxième commande spécifique à dB Start, Compress, permet de compacter les fichiers de données au format dBase. L'opération peut concerner un fichier particulier ou un ensemble de fichiers. A titre d'indication, sachez que dB Start réduit un fichier de 360 Ko à moins de 100 Ko. Une option donne une indication quant au temps estimé pour la décompression. Transfert gère, quant à elle, les transferts de fichiers de PC à PC. Les échanges se font via les ports séries des ordinateurs, à la vitesse de 115 200 bauds. Les fichiers traités peuvent être compressés avant la transmission, automatiquement ou à la suite d'un ordre explicite. La dernière commande originale de dB Start, Emet, prend en charge la transmission de données par l'entremise d'un modem, compatible à la norme Hayes. Reste à vérifier les caractéristiques et les performances de ce produit. Ce que nous ne manquerons pas de réaliser dans un prochain numéro d'*Action PC*.

DB Start est édité et distribué par la société AT&M depuis le début du mois de septembre.

Service lecteur n° 53

dB Start : un clone de dBase III pour moins de 500 F.



DATABOSS 3 : LAISSEZ TRAVAILLER VOTRE MICRO

Générateur d'applications complet, Databoss 3 produit automatiquement menus, fichiers, avec les éventuels liens entre ces fichiers (les bases de données sont gérées par la méthode du séquentiel indexé ISAM), écrans de saisie, états de sortie, aide contextuelle, et documentation développeur. Databoss 3 génère du code source Pascal (Turbo Pascal versions 5.0 ou 5.5 et Quick Pascal), auquel le programmeur peut accéder directement avant la compilation pour y apporter une amélioration ou une personnalisation.

Un générateur d'applications complet.



Les applications créées comportent toutes les caractéristiques professionnelles attendues : écrans de saisie multifenêtre, saisie sous forme de tableaux, gestion des couleurs, gestion de la souris, etc.

Au prix de Databoss 3, 5 995 FHT, il convient d'ajouter l'achat d'un compilateur. Les développeurs d'applications verticales peuvent se procurer Databoss 3 dès à présent, auprès de la société ATEA.

Service lecteur n° 54





LIVRAISON EXPRESS
POUR FOURNITURES ET ACCESSOIRES

Photos Studio BAZIN

KADDE Informatique s'est depuis longtemps fixé un objectif simple : **LA SATISFACTION TOTALE DU CLIENT.**
Une palette complète de service est donc à votre disposition pour **FACILITER VOTRE CHOIX** : appareils de grande marque - conseil avant achat - démonstration de logiciels - audit des besoins informatiques - et pour **FACILITER VOTRE ACTIVITE** : livraison et installation sur site* - formation - livraison express des fournitures et accessoires - SAV et techniciens de maintenance.
Parce que, pour KADDE, une entreprise qui investit dans l'informatique c'est d'abord une entreprise qui investit dans l'efficacité... en toute sécurité.

**Paris et région parisienne.*

KADDE 12ème 55, rue du Rendez-Vous
75012 PARIS (du lundi au samedi)
Tel. : **43 43 12 12**
KADDE 16ème
47, avenue de Versailles
75016 PARIS (du lundi au samedi)
Tel. : **40 50 76 76**

KADDE INFORMATIQUE

Cerclez info pub 11

WINGZ SOUS WINDOWS 3

Informix commercialise une version pour PC du célèbre tableur-grapheur Wingz, adaptée à l'environnement Windows 3. Elle fait suite aux versions de Wingz pour Macintosh et systèmes Unix.

Wingz suit la voie ouverte par Excel dans le domaine des tableurs grapheurs de présentation.

Seule l'approche varie quelque peu par rapport à Excel. Le principe retenu consiste à disposer des objets sur une feuille de calcul. Ces objets contiennent les données de la feuille, ou des graphiques, et peuvent être positionnés à n'importe quel endroit. On peut ainsi, par exemple, placer à côté du tableau de l'évolution du prix de l'essence, un diagramme qui met en évidence la progression constante de celui-ci. Le graphique adopte l'une des dispositions habituelles : barres, escalier, ligne brisée, mais aussi courbes de niveau, barres ou escalier en trois dimensions, représentation cartésienne ou polaire à l'usage des ingénieurs ; encore n'est-ce là qu'un échantillon des possibilités graphiques de Wingz.

Souhaitez-vous faire apparaître le diagramme précédent dans une bulle sortant de la bouche du directeur de Elf ? Rien de plus simple. Il suffit d'importer dans Wingz une image numérisée de celui-ci, de tirer cinq traits (que les fonctions graphiques intégrées se chargent de lisser pour produire une bulle plus que présentable). Il ne reste plus qu'à y placer le fameux graphe. Pour adjoindre un texte, une fenêtre traitement de textes fera l'affaire. Wingz permet beaucoup plus, grâce à son langage de programmation Hyperscript. Si une fonction manque — par exemple un type de graphe très spécifique qui n'aurait pas été prévu — il est possible de la créer. Ces fonctions utilisateur peuvent être saisies sous forme de texte, comme dans un langage classique, ou composées à partir de fonctions existantes. Il suffit de passer en mode apprentissage dans le menu « macros » pour que toute commande utilisée jusqu'à la fin de la macro soit enregistrée dans un fichier script, sous forme d'un programme Hyperscript.

Tout programme de script pourra aussi être associé à une option de menu, ou à un objet « bouton », placé sur la feuille de calcul. Il sera activé par un « événement », qui peut être le choix d'une option d'un menu, un laps de temps sans déplacer la souris (c'est typiquement cette technique qui fait apparaître les sous-menus), un clic sur un bouton, ou encore un déplacement de la souris (par exemple pour déplacer un curseur et régler une intensité de couleur, ou l'ascenseur pour se déplacer dans la fenêtre). La combinaison scripts - événements est suffisamment puissante pour construire de véritables applications. La première de celles-ci est d'ailleurs le tableur-grapheur Wingz lui-même. N'est-ce pas la meilleure preuve de la puissance d'Hyperscript ?

Service lecteur n° 55

UN VERIFICATEUR D'ORTHOGRAPHE EVOLUE

Non content de débusquer les fautes d'orthographe, Hugo Plus se targue de déceler les erreurs de grammaire. En fait, ce logiciel vérifie un certain nombres d'éléments : concordance entre l'article et le nom, l'article et l'adjectif, le nom et l'adjectif, l'adjectif et le nom, le pronom possessif et le nom, et le pronom possessif et l'adjectif, selon le nombre et le genre. Il vérifie également l'accord des verbes en fonction du nombre et du genre des noms, des pronoms personnels, et débusque les prépositions « à » placées devant les verbes à l'infinitif.

Hugo Plus travaille à partir d'un dictionnaire de 400 000 mots, et des dictionnaires « privés ». La vérification peut porter sur des textes de longueur illimitée. Hugo Plus livre des suggestions pour corriger les fautes décelées. Dans le même esprit, l'utilisateur a accès à un logiciel de conjugaison intégré.

Hugo Plus est compatible avec les versions 4.2, 5.0, et 5.1 de WordPerfect, avec Word 5, Wordstar 4, et plus généralement avec l'ensemble des traitements de textes générant des fichiers en Ascii.

La configuration matérielle requise comporte 640 Ko de mémoire vive, deux unités de disquettes, ou un disque dur, ainsi que le MS-DOS 2.0, ou supérieur. Hugo Plus est édité par la société Softissimo au prix de 990 FHT.

Service lecteur n° 56

DES TABLEAUX POUR FRAMEWORK

Le générateur automatique de tableaux « Tableaux-FW » fonctionne avec les fichiers — textes, feuilles de calculs, et bases de données — de Framework II ou III.

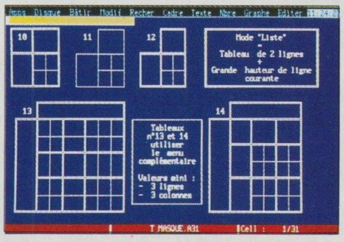
Ce logiciel permet de générer des tableaux vides, qui seront complétés ultérieurement par la saisie directe des données, et d'encadrer automatiquement un fichier existant.

Pour créer un tableau vierge, il suffit de spécifier un certain nombre de paramètres, nom du tableau, type du tableau (choisis parmi les quatorze disponibles), nombre et hauteur des lignes, nombre et largeur des colonnes, etc. Les modèles de tableaux correspondent à des agencements et à des natures d'encadrement (trait simple ou double) différents. La fonction d'encadrement automatique sélectionne le type de tableau adéquat, en fonction de la nature des données à traiter.

Les tableaux créés peuvent bien entendu être modifiés et complétés ultérieurement. Fonctionnant comme un module de Framework, « Tableaux-FW » peut utiliser des fichiers importés de dBase, de Multiplan, etc.

Edité par la société Blanc-Michot Télématique, « Tableaux-FW » est écrit en langage FRED. Il est vendu 1 950 FHT, et ne fonctionne pas indépendamment de Framework.

14 types de tableaux pour les données de Framework.



Service lecteur n° 57

MY TAYLOR IS RICH, BUT MY ENGLISH IS POOR

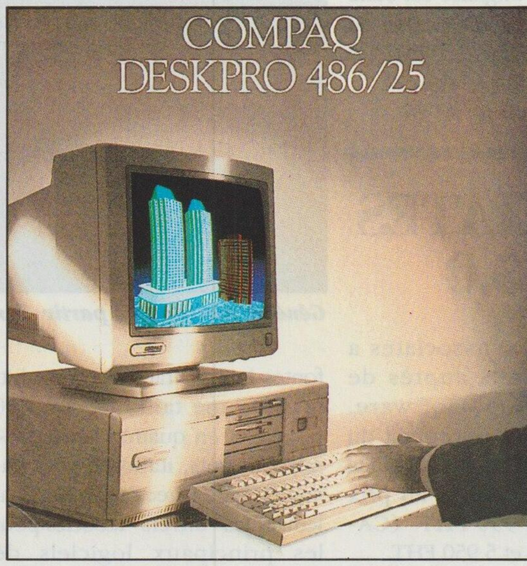
Conjugaison anglaise s'adresse aux élèves des classes de sixième et de cinquième. Il jette les bases de la conjugaison des verbes en langue anglaise. L'habillage graphique des leçons est sobre mais sans austérité. Une partie théorique permet



La conjugaison anglaise livre ses secrets.

COMPAQ

Distributeur agréé



486/25



LTE/286



SYSTEMPRO

PRÊT ET
DÉMONSTRATION
DE TOUTE LA GAMME
AUX GRANDS COMPTES

Tél. 45.66.94.69

DISPONIBILITÉ ET LOCATION IMMÉDIATES

La Dédicace
INFORMATIQUE

173, rue de Vaugirard - 75015 PARIS
Tél. 45.66.94.69
Fax 43.06.35.18 - Télex 202 993

d'aborder les différents temps : présent continu, présent simple, passé simple, passé continu, présent parfait — present perfect —, et futur, dans leurs multiples formes. Des exemples agrémentent la phase d'apprentissage. A tout moment, l'élève a accès à un dictionnaire, de taille très réduite, et à la table des verbes irréguliers. Enfin, le logiciel teste le niveau de compréhension de l'utilisateur au cours de sessions d'exercices, sanctionnées par une note sur 20.

Conjugaison anglaise est édité par Micro C. Il est disponible en version 5"1/4 (prix : 220 FHT) et 3"1/2 (240 FHT).

Service lecteur n° 58

COMPUTER ASSOCIATES ET WINDOWS 3.0

En dépit de nos efforts, il apparaît que le tableau récapitulatif de notre dossier sur Windows 3.0 (*cf.* le numéro 5 d'*Action PC*) péchait par omission. Ainsi, nous avions passé sous silence les efforts de la société Computer Associates en faveur de l'environnement graphique de Microsoft.

Volonté de promotion qui se traduit, dans un premier temps, par la présentation des versions spécifiques pour Windows 3.0 des logiciels CA-Cricket Graph et CA-Cricket Presents. Il s'agit de logiciels — respectivement grapheur et logiciel de PAO — dont Computer Associates a acquis les droits auprès de Xerox Desktop Software. Disponibles dans le courant du mois de septembre, ces produits seront commercialisés au prix public de 2 950 FHT (CA-Cricket Graph) et 5 950 FHT.

De plus, Computer Associates annonce, pour l'année 1991, des adaptations sous Windows des logiciels Textor (traitement de textes), SuperDB (gestion multifichier avec langage de programmation), et SuperProject (gestion de projets et de ressources).

Service lecteur n° 59

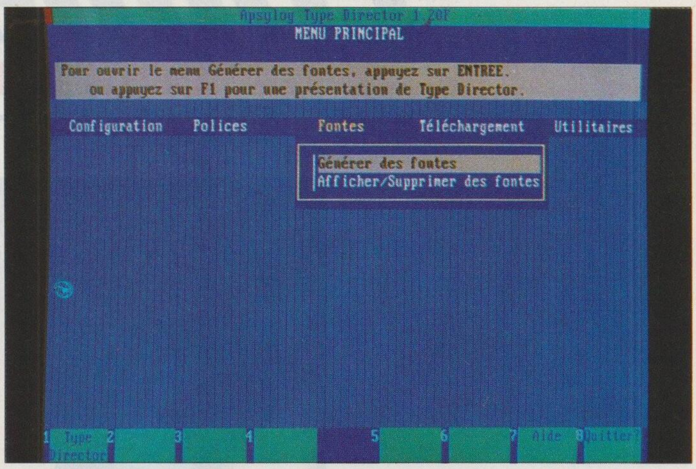
BORLAND CHEZ LES SOVIETS

Borland poussera-t-il la concurrence avec le premier éditeur informatique américain (Microsoft) jusqu'à devenir le premier éditeur soviétique de logiciels ? C'est la question que l'on peut se poser aujourd'hui. L'URSS est, en effet, un marché rêvé qui commence à s'ouvrir et que de nombreuses sociétés tentent d'envahir. Parmi elles, Borland qui vient de sceller un accord avec la société Interquadro (joint-venture réunissant des capitaux français, italiens et soviétiques). Interquadro distribue des produits informatiques à travers toute l'Union soviétique. Si les prévisions gouvernementales s'avèrent exactes, c'est un marché de 700 000 nouveaux PC qu'il faudra installer d'ici à la fin 1991... et équiper en logiciels de bureautique. Les experts prévoient même un parc de 25 millions de machines à la fin de la décennie. Les versions soviétiques de Quattro Pro, Sprint, ou autres Paradox, ont de beaux jours devant elles !

Service lecteur n° 60

DES FONTES SINON RIEN

Grâce au logiciel Apsylog Type Director, les polices Agfa Compugraphics sont dorénavant disponibles sur PC. Le principe de Type Director consiste à générer, à partir de polices vectorielles, des fontes imprimantes et écrans bitmap (d'une taille de 4 à 200 points, et en quatre « versions », normal, gras, italique, gras-italique). Type Director produit également les « drivers » associés pour les principaux logiciels de bureautique du marché : Word, Wordperfect, Ventura, etc. Type Director coûte 990 FHT, ce qui est également le prix de chacune des quelques 40 packs de polices Agfa Compugraphics disponibles.



Générer des fontes à partir d'une simple commande du menu.

Service lecteur n° 61

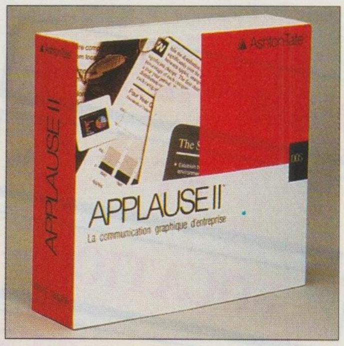
DONNEZ DU RELIEF A VOS CHIFFRES

Applause II insuffle de la vigueur aux rapports et aux bilans. Cet outil de création graphique, permet de concevoir des tableaux et des graphes à partir de fichiers de données aux formats DBF (dBase), WKS et WK1 (Lotus 1-2-3), Ascii, DIF, Sylk, etc. Une option de « lien dynamique » assure la mise à jour des graphiques en cas de modification des données.

Applause II, conçu par Ashton-Tate, génère 37 types de graphes : histogrammes, barres, lignes, tableaux, points, effets 3D, etc. Le logiciel possède huit polices bitstream en standard, utilisées pour composer les légendes et les commentaires. Applause II permet d'importer des images aux formats TIF, GIF, ou PCX, et de les intégrer aux documents créés, en toile de fond ou en surimpression. Le résultat peut être apprécié grâce au mode de prévisualisation WYSIWYG, avant que de lancer l'impression. Applause II est vendu 3 950 FHT par la Commande Electronique. Son installation requiert 512 Ko de mémoire vive. Le logiciel gère les principaux modèles d'imprimantes disponibles, ainsi que de nombreux systèmes d'enregistrement de diapositives et autres projecteurs vidéo.

Service lecteur n° 62

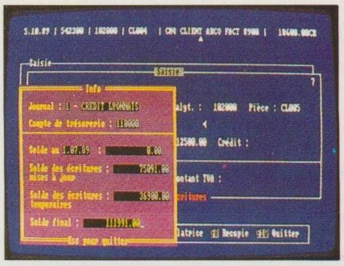
Le graphisme au service de la communication d'entreprise.



BACCARA,

LA COMPTABILITE N'EST PAS UN JEU

La comptabilité générale des professions libérales constitue le domaine de prédilection du logiciel Baccara. La version 2.0 de ce logiciel, édité par la société Trèfle Rouge, gère les opérations



Baccara génère de multiples écrans de saisie des journaux.

les plus courantes. Qu'il s'agisse des encaissements/décaissements, de la gestion des journaux (cinq journaux de trésorerie et un journal d'opérations diverses), de la saisie des écritures, ou de la tenue de la trésorerie en temps réel. Baccara 2.0 permet également de gérer les immobilisations, les démarches fiscales, et peut également se prêter à des manipulations de comptabilité analytique.

Baccara dispose d'une aide permanente et contextuelle, ainsi que d'une interface utilisateur conviviale. Le logiciel nécessite 640 Ko de mémoire vive, un disque dur, et la version 2.0 du DOS, ou supérieure. Baccara 2.0 est vendu 1 650 FHT.

Service lecteur n° 63

MICROLAND MAESTRIA SOUS WINDOWS

L'heure est aux logiciels compatibles avec Windows 3. Le domaine des progiciels de gestion suit le mouvement, et Microland propose depuis le début du mois de septembre une version sous Windows de Maestria.

Cet événement s'accompagne de la récriture complète du produit. Aux fonctionnalités de la version DOS 2.0, Maestria pour Windows ajoute la convivialité de l'interface de Windows, et un certain nombre d'aménagements. L'ensemble des opérations géré par cet outil de gestion comptable et analytique multisociété est accessibles depuis les menus déroulants, par l'intermédiaire de la souris. L'utilisateur peut définir un nombre illimité de masques de saisie des écritures comptables, ouvrir simultanément plusieurs journaux, et consulter rapidement l'historique d'un compte. Une fonction permet d'obtenir des plans prévisionnels.

Parmi les originalités de Microland Maestria, figurent la possibilité de gérer des comptes multidevises, ainsi que l'existence d'un système de protection par mot de passe à plusieurs niveaux. Cette version pour Windows reprend trait pour trait les caractéristiques et l'interface utilisateur de Maestria Mac. Elle peut donc être utilisée sans problème dans le cadre d'un réseau hétérogène PC-Macintosh.

Microland Maestria s'adresse aux PME-PMI, employant jusqu'à deux cents personnes. Son exploitation nécessite 640 Ko de mémoire vive, la version 3.00 ou supérieure du DOS, et bien sûr Windows 3.

Microland commercialise Maestria au prix de 11 850 FHT, et propose des conditions exceptionnelles aux utilisateurs de logiciels de comptabilité. Ainsi, et jusqu'au 31 décembre, ceux-ci peuvent se procurer Meastria pour Windows au prix de 4 950 FHT si le logiciel précédent est un produit Microland (Maestria 2, ou Crocus), ou 6 950 FHT, si tel n'est pas le cas.

Service lecteur n° 64

RECONNAISSANCE DE L'OCR

Conçu par la société Caere Corp. — éditrice du logiciel d'OCR Omnipage —, Parallel Reader est un système complet de reconnaissance de page. Il intègre, dans un sobre boîtier noir, un processeur 386sx à 16 MHz, 1 Mo de mémoire vive, un lecteur de disquettes au format 5" 1/4 de 1,2 Mo, un disque dur de 40 Mo, un contrôleur vidéo VGA, la version 4.0 de MS-DOS, une version runtime de Windows 2.03, ainsi que la partie logicielle du système. Quatre de ses connecteurs accueillent les cartes coprocesseurs Parallel Reader. Un port est réservé à la carte d'interface du scanner, non fournie.

Parallel Reader est compatible avec les numériseurs HP ScanJet Plus, Microtek MS-II, et Fujitsu 309x. Il peut se connecter aux réseaux les plus répandus : Novell Netware, 3Com, etc.

Au chapitre des performances, Parallel Reader traite jusqu'à 2 500 mots à la minute, et peut numériser et reconnaître quatre pages simultanément. La technologie Anyfont, développée par Caere Corp., permet de lire les polices de caractères d'une taille de 6 à 72 points. Elle reconnaît également 11 alphabets différents, et leurs particularités (minuscules accentuées, etc.). Le système Parallel Reader déchiffre encore les attributs du texte (souligné, gras, italique), les impressions laser, les compositions, les caractères à espacements proportionnels ou fixes, le multicolonnage. Enfin, il sépare les textes et les graphiques.

Parallel Reader est distribué en France par Softmart, importateur exclusif d'Omnipage, au prix de 96 500 FHT.

Service lecteur n° 65

Derrière ce boîtier noir se cache une station d'OCR.



L3S SIMPLIFIE LE CABLAGE

Le logiciel CSM (Cabling System Manager) de L3S facilite les opérations d'agencement et de gestion des locaux. CSM permet de visualiser le réseau de câbles, d'identifier les liaisons existantes, de reconfigurer le système, en tenant compte des contraintes du réseau. Dans la pratique, CSM gère une base de données décrivant tous les éléments du réseau : câbles, répartiteurs, boîtiers de raccordement, interfaces, et leurs caractéristiques (éléments précâblés, montés, hors service) cela aussi bien pour le téléphone ou le télex que pour l'informatique et la bureautique. Un système de menus déroulants, de zooms, et de représentations graphiques permet d'analyser précisément le comportement du réseau. CSM fonctionne sur PC, PS ou compatible, sous MS-DOS 3.3 et suivants.

Service lecteur n° 66

VOUS AVEZ DIT SPECIALISE ?

Si CadPipe ne s'adresse pas à un vaste public, en revanche ce logiciel de CAO comblera les professionnels de la tuyauterie. CadPipe permet en effet l'étude et la conception de réseaux de tuyauteries. CadPipe utilise le logiciel AutoCAD comme pilote graphique.

Il se compose de quatre modules : Pid, création de schémas de circulation des fluides (13 000 FHT), Forms, générateur de documents d'ingénierie (6 000 FHT), Ortho, vues en plan de tuyauteries (24 000 FHT), et Iso, isométries et nomenclatures (24 000 FHT).

Le produit complet, distribué par CEFI Technologies, est disponible en version française ou américaine, au prix de 53 000 FHT.

Service lecteur n° 67

POSTSCRIPT LEVEL 2

Cinq ans après l'introduction du langage de description de page PostScript, Adobe en dévoile la première évolution. Cette politique a été jugée plus rationnelle que de multiples et fréquentes améliorations, par les dirigeants d'Adobe.

PostScript 2 s'inscrit dans la lignée de ce langage, leader sur le marché, en lui apportant de nombreuses améliorations. Ainsi, il intègre certains opérateurs d'affichage des graphiques et des textes déjà éprouvés dans le logiciel Display PostScript. De même la compression des fichiers est optimisée, de manière à réduire les temps de transmission, point faible du système. Adobe a également travaillé au niveau de l'interpréteur du langage pour diminuer les temps de création des caractères et améliorer la qualité des polices de petites tailles, et des textes en basse résolution.

PostScript 2 permet à l'utilisateur de définir un fond de page virtuel. Ainsi, d'un document à l'autre, seules les informations modifiées sont de nouveaux traitées et interprétées. Notons encore des améliorations quant à la gestion des imprimantes (alimentation, format du papier, agrafage, etc.), le traitement de la couleur.

PostScript Level 2 assure une compatibilité totale avec le PostScript « traditionnel ». Disponible depuis début juillet sous licence OEM, les premières imprimantes basées sur le langage PostScript 2 n'apparaîtront pas avant le début de l'année 1991 sur le marché.

Service lecteur n° 68

WINDOWS SOUS LES ORDRES DU STARTER

La société Bitstream a dévoilé le Windows Starter Kit, logiciel de compilation et de téléchargement de polices de caractères. Fonctionnant sous Windows, versions 2.XX et 3.0, il remplace le logiciel Fontware 2.0.

Le Windows Starter Kit dispose de divers pilotes d'imprimantes : PostScript, HP LaserJet, DeskJet, Canon LBP 8II, IBM Proprinter, etc. Livré avec deux fontes en standard (Swiss et Dutch), le kit peut utiliser les fontes Bitstream (48 « packages » différents). L'utilisateur peut également générer ses propres jeux de caractères, et les utiliser dans le cadre d'applications. Le Windows Starter Kit est distribué par la société ISE-Cegos, au prix de 600 FHT.

Service lecteur n° 69

LE REPONDEUR TELEPHONIQUE DU FUTUR

Auralog commercialise un serveur vocal interactif, une sorte de super-répondeur « intelligent » : AURA-SERV. Ce serveur parle à son interlocuteur, et réagit en fonction de la réponse. La réponse est limitée à un mot ou à une phrase courte, parmi un choix proposé par le serveur, un menu téléphonique en quelque sorte. Le mode d'emploi de ce type de serveur est donc on ne peut plus simple : composer un numéro, écouter, répondre.

Le Comité Régional du Tourisme Alpes Savoie Mont-Blanc à Aix-les-Bains vient d'acquérir un serveur Auralog pour répondre aux demandes d'information des touristes et rediriger leurs appels vers des prestataires : hôtels, centrales de réservation, offices du tourisme, etc. Un serveur monovoie, avec une arborescence simple et une vingtaine de mots clés coûte quand même la bagatelle de 100 000 F. Beau répondeur !

Service lecteurt n° 70

DIALOGUE TERM A TERM

TERM est un logiciel de communication programmable asynchrone qui a l'agréable particularité de fonctionner de façon identique sous MS-DOS, UNIX, XENIX, CTOS, BTOS, CP/M, Macintosh et sur plus de cent machines différentes. TERM permet l'utilisation d'un PC à distance, par l'intermédiaire d'un modem, ainsi que l'émulation de divers terminaux. Il dispose de nombreux protocoles (Kermit, XModem, YModem, WTERMC, ou TERM — le protocole interne de TERM qui effectue une compression des données avant transfert), de la numérotation multiple, ou encore de l'accès multimodem. Un langage de scripts puissant est intégré au produit. Pour personnaliser ses applications, l'utilisateur peut manipuler fenêtres, menus, boucles, accès à des fichiers, divers types de données. Les scripts sont entièrement compatibles d'un système d'exploitation à l'autre. TermNET comporte les mêmes fonctionnalités dans le cadre d'une utilisation en réseau.

Service lecteur n° 71

UN PC PARLE AUX PC

La multiplication des modems explique la prolifération de logiciels de télémaintenance tels que TelySPC. Ce dernier permet de relier deux PC, et d'effectuer des transferts d'information entre les deux postes à des vitesses de 75 à 19 200 bauds, selon le mode de transmission choisi. TelySPC reconnaît automatiquement la configuration matérielle du correspondant. La relation maître-esclave peut être aisément inversée, selon les besoins de l'utilisateur. TelySPC occupe 50 Ko en mémoire. Il dispose également de fonctions de protection par mot de passe. Edité par Fidis Télématique, TelySPC est vendu 950 FHT.

Service lecteur n° 72

DU NOUVEAU POUR NOVELL

L'ultime évolution du logiciel LANsmart est apparue au début de l'été.

Conçu par iD-BiT, cet utilitaire s'utilise dans le cadre d'un réseau Novell Netware. Il permet de définir de nouveaux serveurs, tant d'impression que de fichiers, se substituant alors au réseau, en le soulageant d'un certain nombre de tâches. Grâce aux aptitudes de LANsmart, les ressources du réseau peuvent alors être accrues — définition de nouveaux serveurs d'impression ou de fichiers —, et les risques de saturation écartés.

LANsmart est proposé au prix de 6 990 FHT.

Service lecteur n° 73


TOSHIBA
Distributeur agréé
Papman T 1000/SE
T 1200 HB/SX
PRÊT ET
DÉMONSTRATION
DE TOUTE LA GAMME
AUX GRANDS COMPTES
Tél. 45.66.94.69
T 1600
T 3100 e/SX
T 3200 SX
T 5100
T 5200
DISPONIBILITÉ
ET
LOCATION
IMMÉDIATES
La Dédicace
INFORMATIQUE
173, rue de Vaugirard - 75015 PARIS
Tél. 45.66.94.69
Fax 43.06.35.18 - Télex 202 993

NOUVEAUX DRIVERS DE RESEAUX POUR WINDOWS 3.0

Les réseaux locaux d'entreprises disposent désormais d'une interface complète sous Windows 3, grâce aux nouveaux drivers d'Ungermann-Bass. Avec « NET/One TCP for Windows 3.0 », l'utilisateur accède à des ressources et applications réseau en mode protégé TCP/IP, TELNET et NETBIOS. Avec la combinaison Driver Net/One et Windows 3.0, l'utilisateur peut simultanément charger et utiliser une application sur site central dans l'une des fenêtres, accéder et manipuler les données d'un serveur dans une autre, et imprimer sur n'importe quelle imprimante du réseau.

Service lecteur n° 74

WHAT'S ON VERSION 2

La société Allphase Informatique annonce la disponibilité de la nouvelle version de What's On. Utilitaire permettant de gérer son disque dur, What's On s'affranchit de la notion de répertoire et de partition disque. What's On est très simple d'utilisation, la prise en main se fait en quelques minutes seulement. Avec cette nouvelle version il est maintenant possible (en plus des traditionnelles fonctions de recherche de chaîne de caractères, cryptage de fichier, changement d'attributs, de la date et de l'heure d'un fichier) de trier des répertoire par nombre de fichiers, de définir des fichiers à ignorer lors des recherches de chaînes, de paramètrer totalement les couleurs d'affichage, etc. La sécurité des manipulations a été également améliorée. Sans être révolutionnaire What's On est un outil très pratique pour partir à la découverte de son disque dur et permettre d'en optimiser la gestion.

Service lecteur n° 75

QRAM, LES 8088 NE RAMENT PLUS

QRam est un gestionnaire de la mémoire dédié aux PC munis de processeurs 8088 ou 8086. Il permet d'accroître les performances de ces systèmes, en optimisant la gestion de la mémoire EMS et de la mémoire vidéo. QRam est compatible avec la mémoire EMS 4.0, EEMS. Le principe est des plus classiques : QRam installe en mémoire EMS les logiciels résidants, et les drivers, libérant ainsi de 30 Ko à 130 Ko de mémoire. QRam peut également utiliser la mémoire vidéo si celle-ci n'est pas exploitée. QRam est un produit Quaterdeck (éditeur de Qemm 386). Innosoft assure sa diffusion en France, au prix de 990 FHT. Dès le mois d'octobre, une version française se substituera à la version anglaise actuellement commercialisée.

Service lecteur n° 76

PC-KWIK POWER PAC ACCELERE LES PC

Sofmart est le distributeur exclusif en France de PC-Kwik Power Pak. Ce produit réunit cinq utilitaires permettant d'accélérer les accès aux principaux périphériques d'un micro-ordinateur : un cache mémoire, jusqu'à 512 Ko en mémoire conventionnelle et 16 Mo en mémoire étendue ; un spooler d'imprimante capable de gérer deux imprimantes et de compresser les données pour préserver l'espace mémoire du buffer ; deux disques virtuels (RAM disk) extensibles jusqu'à 16 Mo ; un accélérateur d'écran qui intègre une fonction de scrolling avant et arrière sous DOS ; et enfin un accélérateur de clavier. Ces utilitaires sont installés simultanément et partagent le cache alloué.

Service lecteur n° 77

UNIFY UN LANGAGE D'INTERROGATION SQL/A INTEGRE POUR COBOL

Unify Corporation fabricant d'outils logiciels de développements d'applications sous système Unix, créé en 1980 aux Etats-Unis, annonce la disponibilité du langage ESQL/A pour Cobol. ESQL/A a été conçu comme une extension du langage Cobol. Il permet à l'intérieur d'un programme Cobol d'intégrer des requêtes SQL à la base de données Unify 2000. Ce type d'outils, existe également chez Informix, Oracle, Ingres — pour ne citer que les plus connus. Ils offrent l'immense avantage pour les développeurs d'applications de rester cohérents avec les centaines de milliers de lignes de code Cobol qui ont pu être écrites à travers le monde depuis maintenant plus de trente ans, tout en y apportant les avantages de facilité, d'efficacité et de standardisation offerts par SQL et sa représentation de données.

Service lecteur n° 78

TURBO C++ EN FRANÇAIS !

Turbo C++ est disponible en langue française depuis le mois d'août. N'ayant pu disposer du produit pour le bouclage de ce numéro, espérons que les manuels ont été correctement traduits et qu'ils sont encore au nombre de quatre (habituellement, il y en a au moins un qui disparaît lors de la traduction). On souhaite aussi que la partie initiation au langage C ait été enrichie. A l'occasion de cette annonce, on laissait espérer la disponibilité pour les mois de septembre ou d'octobre de Paradox Engine dans les langages Turbo-Pascal et Turbo C++. Rappelons que Paradox Engine (à l'heure actuelle uniquement disponible en C) est une bibliothèque de procédures et de fonctions permettant d'interfacer ses propres applications avec le SGBD Paradox.

Service lecteur n° 79

MODULA 2, VERSION 2

Après le succès européen et américain de la première version du langage TopSpeed Modula 2, JPI propose une nouvelle version de son langage.

L'environnement de développement possède les mêmes caractéristiques que celui du C (neuf fenêtres, table Ascii, calculette, macros, etc.), mais là n'est pas la principale amélioration. Modula 2 dispose maintenant de six modèles mémoires différents, dont un réservé au

développement sous OS/2. Les paramètres des procédures et fonctions sont de préférence passés par registre, de nouveaux modules permettent l'intégration du support de la souris, la redirection des fichiers d'entrée/sortie, et l'accès aux bibliothèques C. Les modules existants ont aussi été étendus, et notamment en ce qui concerne le graphisme (heureusement !), et les chaînes de caractères. La gestion des overlays est automatique (on pense à VROOM...), le code source des bibliothèques est fourni dans la version « professionnelle » du langage. Des exemples d'applications sous Windows sont donnés, ainsi qu'un module permettant le support de l'intégrateur graphique de Microsoft. Le Modula 2 est maintenant orienté objets (mêmes spécificités que le Turbo-Pascal 5.5).

L'optimiseur de code devient commun à tous les langages TopSpeed. L'environnement de développement est également le même - ce qui permet, entre autres choses, de lier plusieurs fichiers provenant de divers langages dans la même application. Seuls les compilateurs sont, bien évidemment, spécifiques à chaque langage. La sécurité a été accrue avec la disponibilité de « pragmas de compilation », qui sont eux aussi utilisés pour les autres langages. Le Modula 2 version 2.0 n'a pas changé de prix par rapport à la version précédente. Le produit est disponible dans trois configurations : l'édition « standard » contient, outre le compilateur et les modules, le debugger interactif et l'interface avec Windows ; l'édition « étendue » comprend aussi les sources des modules, un assembleur, un désassembleur, un Profiler, et le support pour overlays automatiques (DLL) ; l'édition OS/2 correspond à l'édition DOS étendue, mais avec un compilateur spécifique à OS/2 qui permet le développement en mode protégé. Nous reviendrons, dans notre prochain numéro, sur cette version 2 de TopSpeed Modula 2.

Service lecteur n° 80

CONFERENCE DES DEVELOPPEURS NANTUCKET EN FRANCE

Cette manifestation organisée par PC Technologie, distributeur/importateur de Nantucket en France, qui s'est tenue les 29, 30 et 31 mai derniers, a été l'occasion de rencontres et d'échanges de points de vues entre développeurs et personnalités de Nantucket. Société fondée en Californie en 1984, Nantucket est l'éditeur du SGBD pour micro Clipper dont la dernière version, estampillée 5.0, a été présentée au cours de ces journées. Ce clone le plus connu de dBase, ne cherche plus aujourd'hui la comparaison systématique avec son grand frère et concurrent mais semble préférer dorénavant voler de ses propres ailes.

Service lecteur n° 81

PROLOGUE 3 SUR COMPAQ SYSTEMPRO

MS-DOS, OS/2 et Unix ne sont pas les seuls systèmes d'exploitation. Prologue, développé à l'origine par Bull Micral, puis par Prologue s.a., est le leader européen du multiposte. La version de Prologue 3 pour Compaq Systempro utilise pleinement les capacités exceptionnelles de cette machine, notamment grâce à l'emploi de requêtes envoyées au contrôleur multidisque intelligent. Celui-ci sait minimiser les déplacements de la tête et réduit ainsi considérablement les temps d'attente des utilisateurs.

Service lecteur n° 82

■ Epître 1.0 est un logiciel de bureautique intégré fonctionnant sous Unix. Deux modules, tableur et traitement de textes, sont actuellement disponibles auprès de la société Isix, au prix de 15 000 FHT.

Service lecteur n° 83

■ La version 1.1 de Personal Machinist, logiciel de CFAO, a été présentée fin juillet par Prime-CBU.

Service lecteur n° 84

■ Check It bientôt disponible en France, grâce à l'accord conclu entre l'éditeur américain de cet utilitaire de diagnostic, TouchStone Software, et Softpublishing. Disponibilité annoncée pour la fin de l'année.

Service lecteur n° 85

■ Altelex est un logiciel de communication bilingue. Couplé à un boîtier Textmaster, il permet de diffuser des télex à partir d'un PC. Altelex est édité par ALT Computer.

Service lecteur n° 86

■ La version française de PC-Tools est maintenant disponible. Les quelques problèmes que nous avions signalés lors du banc d'essai du précédent numéro ont bien entendu été corrigés.

Service lecteur n° 87

■ Le Logiciel de création de formulaires Jetform 2.2 répond aux spécifications de Windows 3. Cette version est disponible depuis le mois de septembre auprès de Lazerprint, au prix de 7 500 FHT.

Service lecteur n° 88

■ Plume 3 Science est un traitement de textes scientifique entièrement graphique et WYSIWYG. Il possède une bibliothèque de symboles mathématiques, ainsi que des caractères spéciaux utilisés dans certaines formules chimiques. Plume 3 Science est édité par Andiade Systèmes.

Service lecteur n° 89

■ Question, logiciel de traitement d'enquêtes, génère tableaux, graphiques, et données statistiques. La distribution de Question est assurée par AID/MLI (à partir de 8 000 FHT pour le module de base).

Service lecteur n° 90

■ Windows 3 équipera dorénavant en standard certains modèles de la gamme Bravo des micro-ordinateurs AST Research. Cette offre concerne en particulier des micro-ordinateurs à base de processeurs 286 et 386sx.

Service lecteur n° 91

■ La version française de Fox Pro Réseau 1.01 est disponible depuis le mois d'août. AB Soft en assure la distribution : 17 950 FHT pour la version de base, 23 950 FHT pour le kit développeur (version réseau plus module runtime illimité). La mise à jour est gratuite pour les possesseurs de la version américaine du SGBD relationnel de Fox Software.

Service lecteur n° 92

■ DeSribe Word Publisher 1.1 est un traitement de textes sous OS/2 Presentation Manager. Doté de fonctions de mise en page, Word Publisher est commercialisé par Blue Link au prix de 5 950 FHT.

Service lecteur n° 93

■ Facturation enrichit la gamme des logiciels de gestion Arc-en-Ciel Elite. Il peut communiquer avec l'ensemble des produits de cette famille, en particulier Comptabilité.

Service lecteur n° 94

■ Base de données relationnelle, Omnis 5 fonctionne sous environnement Windows 3. La société Aware assure la distribution d'Omnis 5 (6 950 FHT).

Service lecteur n° 95

■ Merisel est né de l'union de Softsel et Microamerica, pour former le premier distributeur de produits micro-informatiques.

Service lecteur n° 96

■ Le système d'exploitation DR DOS 5.0 équipera certains modèles des gammes Glaad, Tetratek, Maine Informatique.

Service lecteur n° 97

■ RX Image est un logiciel de mise à la couleur, gérant 256 couleurs parmi une palette de 16 millions de teintes. Il possède une option de séparation des couleurs pour une impression en quadrichromie. RX Image traite les fichiers aux formats TIFF, SLD (AutoCAD), AutoShade

(RND), et GIF (Animator). RX Image est distribué par ISTC (4 950 FHT).

Service lecteur n° 98

■ Pafec distribue Duct 386, un logiciel de CFAO (Conception et Fabrication assistée par ordinateur) réservé aux micro-ordinateurs 386.

Service lecteur n° 99

■ Rescue 1.00 permet de diagnostiquer les applications standard et de repérer d'éventuels virus. Toute anomalie (fichiers perdus, modifiés, parasités, etc.) est immédiatement signalée. Rescue 1.00 est édité par la société Log'Up (1 400 FHT).

Service lecteur n° 100

■ Le logiciel de communication Smartcom III est doté des émulations Minitel, VT 100 et VT 102. Il gère les protocoles de transfert Xmodem, Ymodem et Kermit. Métrologie en assure la distribution (1 700 FHT).

Service lecteur n° 101

■ Project pour Windows est disponible en version française. Le nouveau gestionnaire de projets de Microsoft est proposé à 6 990 FHT.

Service lecteur n° 102

ACTUALITES LIVRES

L'INDISPENSABLE POUR LASERJET

Ce petit ouvrage de C. Centner aux éditions Marabout (224 pages, 55 F, format de poche) constitue une excellente approche des possibilités (et des limites) de votre imprimante Laser HP (Laserjet Plus et série II) ou compatible. Après avoir décrit le fonctionnement du mode d'impression laser, le tableau de commande de la Laserjet, les problèmes de connexion, l'auteur fait le jour sur de nombreux secrets de cette imprimante : les polices en cartouches, les polices logicielles, le langage PCL, les possibilités d'édition graphique. A conseiller, donc, avec ferveur, à tous ceux qui souhaitent tirer le maximum de leur Laserjet.

Service lecteur n° 103

GUIDE PSI DU MATERIEL DES PC, PS, ET COMPATIBLES

Si la plupart des livres traitant de micro-informatique font une large part à la programmation, les aspects matériels du PC sont généralement oubliés. Que sait-on, au fond, de la carte mère, du fonctionnement de la mémoire vive (RAM) ou du bus système ? C'est pourtant là que tout commence, et ce n'est pas la moindre des qualités du livre de Winn Rosch (traduit par Catherine Grenet) que de nous le rappeler. L'auteur adopte le point de vue du bricoleur, et nous apprend à démonter et remonter tous les éléments d'un PC. A la fin du livre, le lecteur est largement en mesure de monter lui-même son micro-ordinateur à partir d'éléments achetés séparément. Surtout, il comprendra bien mieux son fonctionnement, et pourra s'attaquer sans crainte à la programmation du matériel. C'est du reste un des buts avoués de l'auteur que de démythifier les micro-ordinateurs, qui ne sont jamais que des machines. Les nombreux schémas et tableaux, précis et bien conçus aideront la compréhension, de même que l'annexe qui donne la signification de toutes les abréviations employées par les constructeurs. L'index fourni permettra de s'y retrouver sans difficulté. Il s'agit donc là d'un bon ouvrage pour un prix (335 F) encore acceptable en regard de la pagination (648 pages).

Service lecteur n° 104

LES NORTON UTILITIES EN ACTION

Contrairement à ce que laisse entendre la couverture, cet ouvrage n'a pas été écrit par Peter Norton, mais par Rob Krumm (et traduit par Jean-Christophe et Jean-Marie Krust). Il n'a d'ailleurs, ni l'intérêt, ni les qualités d'un livre de Norton. Ce guide — presque exhaustif — des logiciels de Peter Norton Computing (les Norton Utilities, mais aussi Norton Commander, Norton Editor, et les Norton Guides) voudrait bien jouer le rôle d'un manuel. Mais voilà, il est beaucoup trop éparpillé et obscur pour cela. Plan incompréhensible, texte relativement peu riche, présentation désagréable, tels sont les moindres de ses défauts. Ce n'est vraiment pas le livre à conseiller en remplacement des manuels, surtout au prix élevé de 345 F (aux éditions P.S.I).

Service lecteur n° 105

WORD 4 ET 5 AVANCE : LES MACROCOMMANDES

Word perdrait une bonne partie de son intérêt s'il ne disposait de nombreuses possibilités d'automatisation, et notamment de macrocommandes évoluées. Celles-ci ne sont malheureusement ni simples à mettre en œuvre, ni faciles à comprendre si l'on ne possède que le manuel de Microsoft. On comprend mieux l'ambition de ce petit guide écrit par Catherine de Schaetzen dans la collection Marabout : faire enfin comprendre aux utilisateurs pourquoi Word est un bon traitement de textes en les initiant au monde des macros. De simple utilisateur, on passe en douceur au rang de superutilisateur. Tout est expliqué et détaillé de façon simple et agréable, et de nombreux « programmes » utilitaires ou de jeux sont listés en fin d'ouvrage. Voilà, pour moins de 50 F, un excellent achat.

Service lecteur n° 106

MAITRISER MS-DOS 4.0

La version 4.0 de MS-DOS semble inspirer de nombreux auteurs de guides. Cet ouvrage n'est ni plus mauvais ni plus attrayant qu'un autre, et a pour but de vous apprendre la syntaxe de « Format », « Append », et autres « Type » divers. Vous aurez même la joie d'apprendre une fois de plus comment l'on écrit un fichier batch (.BAT) ! Passionnant, non ? Merci à Michel Laurent et aux éditions Eyrolles pour ce moment de divertissement.

Service lecteur n° 107

L'INDISPENSABLE POUR AUTOCAD

Bien que présentant quelques défauts, eu égard à son prix, le logiciel AutoCAD avec plus de 300 000 licences vendues dans le monde, est bien le leader incontesté des systèmes de CAO (Conception Assistée par Ordinateur). Cet ouvrage, paru aux

DAEWOO

PROMO. T.T.C.
☎ 45 66 94 69

DAEWOO DPC/386S

80386SX à 16 MHz, 1 lecteur 5,25'' de 1,2 Mo et 1 lecteur 3,5'' de 1,44 Mo, 4 connecteurs 16 bits, 2 connecteurs 8 bits, adaptateur vidéo VGA/EGA/CGA/MDA/Hercules, 2 ports série, 1 port parallèle, 1 port souris. MS-DOS 4.01, GW BASIC.

Avec 2 Mo de RAM

113020450	Sans disque dur	**12 000 F**
113022450	Disque dur de 40 Mo	**15 000 F**
113024450	Disque dur de 80 Mo	**17 250 F**
113025450	Disque dur de 120 Mo	**18 750 F**
113026450	Disque dur de 160 Mo	**21 000 F**

Avec 4 Mo de RAM

113020550	Sans disque dur	**15 000 F**
113022550	Disque dur de 40 Mo	**18 000 F**
113024550	Disque dur de 80 Mo	**20 250 F**
113025550	Disque dur de 120 Mo	**21 750 F**
113026550	Disque dur de 160 Mo	**24 000 F**

DAEWOO DPC/286

80286 à 8/12 MHz, 1 Mo de RAM, 1 lecteur 5,25'' de 1,2 Mo et 1 lecteur de 3,5'' de 1,44 Mo, 4 connecteurs 16 bits, 2 connecteurs 8 bits, adaptateur vidéo VGA/EGA/CGA/MDA/Hercules, 1 port série, 1 port parallèle, MS-DOS 4.01, GW BASIC.

112020350	Sans disque dur	**9 000 F**
112022350	Disque dur de 40 Mo	**12 000 F**
112024350	Disque dur de 70 Mo	**14 250 F**
112025350	Disque dur de 120 Mo	**15 750 F**

PORTABLE DAEWOO DLT/386S

80386SX à 16 MHz, 1 lecteur 3,5'' de 1,44 Mo, adaptateur vidéo VGA/EGA/CGA/MDA/Hercules, 1 port parallèle, 1 port série, 1 port souris, 1 port RVB, 1 port lecteur externe, 1 connecteur modem, 1 adaptateur allume cigare, écran LCD rétro-éclairé (640 × 400), MS-DOS 4.01, GW BASIC, poids 5,9 kg. Housse de transport.

Avec 1 Mo de RAM

110322340	Disque dur de 40 Mo	**20 250 F**

Avec 2 Mo de RAM

110322440	Disque dur de 40 Mo	**22 500 F**

DAEWOO DPC/386 - 20

80386 à 8/20 MHz, 1 lecteur 5,25'' de 1,2 Mo et 1 lecteur 3,5'' de 1,44 Mo, 4 connecteurs 16 bits, 2 connecteurs 8 bits, adaptateur vidéo VGA/EGA/CGA/MDA/Hercules, 2 ports série, 1 port parallèle, MS-DOS 4.01, GW BASIC.

Avec 2 Mo de RAM

113120450	Sans disque dur	**18 000 F**
113122450	Disque dur de 40 Mo	**21 000 F**
113124450	Disque dur de 80 Mo	**23 250 F**
113125450	Disque dur de 120 Mo	**24 750 F**
113126450	Disque dur de 160 Mo	**27 000 F**

Avec 4 Mo de RAM

113120550	Sans disque dur	**21 000 F**
113122550	Disque dur de 40 Mo	**24 000 F**
113124550	Disque dur de 80 Mo	**26 250 F**
113125550	Disque dur de 120 Mo	**27 750 F**
113126550	Disque dur de 160 Mo	**30 000 F**

MONITEURS MONOCHROMES

CMM - 210 (12801)	12'' TTL (MDA/Hercules), écran vert Résolution 720 × 350 Fréquence horizontale : 18,4 KHz	**750 F**
CMM - 220 (12811)	12'' VGA monochrome, écran plat, page blanche Résolution : 640 × 480, 400, 350 Fréquence horizontale : 31,5 KHz	**1 050 F**
CMM - 310 (12861)	14'' VGA monochrome, écran plat, page blanche Résolution : 640 × 480, 400, 350 Fréquence horizontale : 31,5 KHz	**1 200 F**
CMM - 320 (12871)	14'' TTL (MDA/Hercules), écran plat, page blanche Résolution : 720 × 350 Fréquence horizontale : 18,4 KHz	**1 000 F**

MONITEURS COULEURS

CMC - 430 (12721)	14'' EGA, Résolution : 640 × 350, 200 Diamètre des pixels : 0,31 mm Fréquence horizontale : 21,85 / 15,75 KHz	**2 950 F**
CMC - 420 (12711)	14'' VGA, Résolution : 640 × 480 Diamètre des pixels : 0,31 mm Fréquence horizontale : 31,5 KHz	**3 000 F**
CMC - 410 (12701)	14'' Multimode, Résolution : 800 × 600 (VGA PLUS) 640 × 480, 400, 350, Diamètre des pixels 0,31 mm Fréquence horizontale : 15,5 - 35 KHz	**3 750 F**

Service lecteurs n° 78 (dernière page)

La Dédicace INFORMATIQUE : 173, rue de Vaugirard 75015 PARIS - **Tél. 45 66 94 69** - Fax. 43 06 35 18 - Télex 202 993

éditions Marabout, servira à la fois de guide pour le débutant et d'aide-mémoire pour l'utilisateur expert d'AutoCAD. Grâce à son format réduit, qui facilite les manipulations et les recherches, de nombreux exemples et copies d'écrans, ainsi que des explications concises, ce livre se révèle un parfait complément des volumineux manuels de référence fournis avec le logiciel. Présentées sur la base de la version 10 pour systèmes MS-DOS, les commandes restent néannmoins valables pour les autres plates-formes matérielles d'AutoCAD comme Macintosh, Sun, Apollo et Dec.

Service lecteur n° 108

MICROSOFT MOUSE PROGRAMMER'S REFERENCE

Si vous ne parvenez pas à programmer une interface utilisant la souris dans votre langage préféré à la fin de ce livre, ce ne sera pas de la faute des auteurs ! Ce véritable guide de référence est maintenant disponible aux éditions Microsoft Press (Penguin Books), après être longtemps resté dans les tiroirs de Microsoft Corp. Il était alors aussi difficile à obtenir que le MS-DOS Programmer's Reference.

Livré avec deux disquettes pleines de programmes utilitaires dédiés à la souris, il donne tous les renseignements utiles à sa gestion dans tous langages (il y a même une annexe concernant les conventions d'appels d'interruptions en Turbo-Pascal !). Après deux parties presque inutiles (historique de la souris Microsoft et programmation par l'intermédiaire de menus), on aborde le cœur du problème. Tout est détaillé, des appels de fonctions aux exemples de programmes d'interface souris, en passant par les spécificités imposées par les modes graphiques de la carte EGA. Les programmes sont présentés en GW-Basic, en Quick-Basic, en Microsoft C et Quick C, en assembleur, en Fortran (!), et en Microsoft Pascal.

Service lecteur n° 109

SYSTEMES D'EXPLOITATION, CONCEPTS ET ALGORITHMES

McGraw-Hill France a fait preuve de courage en publiant dernièrement cet ouvrage : le Tannenbaum (chez InterÉditions) et le Krackowiac (chez Dunod) étaient jusqu'alors les deux seules références de fait sur le sujet.

L'approche adoptée ici se veut fondamentalement différente. Le lecteur lambda n'appréciera guère les nombreuses formules mathématiques et leur représentation sous forme de graphes. Mais voilà, ce livre est plutôt destiné à l'ingénieur et à l'étudiant de troisième cycle en informatique qu'au lecteur lambda (les auteurs, Joffroy Beauquier, et Béatrice Bérard sont professeur et maître de conférence universitaires). Il s'agit en effet plus de théorie que de pratique (Tannenbaum publiait, pour sa part, le source d'un Système Unix amélioré). L'ouvrage s'oriente résolument vers les systèmes multitâches et multiprocesseurs. Si l'on est loin d'une description pratique mettant l'accent sur les systèmes existants (généralement MS-DOS et Unix), tout ce qui concerne les sémaphores, les commutations de tâches et leur synchronisation se trouve expliqué par le menu. Les algorithmes font l'objet d'une étude critique sans concession. Un achat utile si l'on est intéressé par les principes de fonctionnement auxquels obéissent tous les systèmes d'exploitation. Peut-être une acquisition à faire au moment du passage de DOS à OS/2 ? Surtout que le prix (environ 250 F) reste correct.

Service lecteur n° 110

MS-DOS PAPER'S

Si vous ne savez pas ce qu'est la « System File Table », et comment elle s'interface avec les champs réservés du PSP (ndlr : et que la question vous intéresse), alors ce livre est pour vous. La seconde édition des MS-DOS Paper's, parue chez Sams, ne faillit pas à la règle de qualité jusqu'ici observée dans les précédents ouvrages du Waite Group.

Celui-ci vient en complément de MS-DOS Encyclopedia, MS-DOS Programmer's Reference, ou MS-DOS avancé. Contrairement à ces derniers (publiés par Microsoft, Penguin Books, et PSI), il fournit en effet nombre de renseignements sur les interruptions et les fonctions cachées — dites « réservées » — du DOS.

Ce sont surtout la seconde et la troisième partie du livre qui sont les plus intéressantes. Tout vous sera révélé sur les techniques les plus avancées de la programmation sous DOS. Outre l'utilisation des fonctions cachées, vous saurez comment écrire un programme résidant respectant les règles utilisées par les grands éditeurs, comment encrypter des données de manière sûre, écrire un driver (deux exemples sont fournis), ou interfacer vos programmes avec le matériel (carte EGA, port série, EMS). Les programmes sont de très grande qualité, et les textes restent clairs. On reprochera toutefois une première partie inutile (il s'agit d'une visite guidée de MS-DOS, assortie d'articles vantant certains Sharewares vendus par les auteurs !), ainsi que l'emploi abusif du langage C et de l'assembleur. Mais le reste vaut bien le prix de l'ouvrage entier (275 F).

Service lecteur n° 111

AU CŒUR DU DOS,
INITIATION APPROFONDIE AU SYSTEME DOS

Quand on s'appelle Peter Norton et que l'on a déjà publié quatre bons livres sur le PC, sa programmation, l'assembleur, et le disque dur, on pourrait se passer de réécrire un guide de l'utilisateur de MS-DOS, pensent les béotiens. Eh bien, « pas du tout ! » répondent en chœur Norton, Brady Books, et PSI. Si vraiment vous aviez perdu le manuel, et que vous n'ayez pas déjà quelque chose comme le Super MS-DOS de Van Wolwerton (même éditeur), vous pourriez être intéressé par un tel ouvrage, qui a le mérite d'être clair et progressif (on apprend comment écrire le fichier AUTOEXEC.BAT au chapitre 12, p. 153, et comment utiliser EDLIN au chapitre 29 !). Mais à 265 F, on examinera à deux fois ses besoins.

Service lecteur n° 112

RESEAUX, ARCHITECTURES, PROTOCOLES, APPLICATIONS

A peine paru dans sa version américaine (chez Prentice-Hall), ce livre faisait déjà autorité en la matière. Il faut dire que Tannenbaum est doué pour rendre accessible ce qui est compliqué.

L'une des méthodes qu'il emploie pour cela est la présentation « en couches » des différents sujets qu'il aborde. Les réseaux ne font pas exception, et c'est donc sous cet aspect que le lecteur fera leur connaissance. Il apprendra ainsi ce que sont les couches physiques (transmission analogique ou numérique), liaison de données (protocoles et codes correcteurs), réseau (routage et interconnexion), transport (connexion), session (appels de procédures), présentation (compression et encryptage de données), et enfin la couche application (messageries électroniques, etc.). Chaque grande partie s'ouvre sur les problèmes posés, se poursuit par l'étude des algorithmes appropriés, et se termine avec la présentation de solutions mises en œuvre dans divers réseaux existants. A ce sujet, il faut remarquer que les exemples sont pris à la fois en Europe et aux Etats-Unis : on étudiera bien sûr Usenet, mais aussi Transpac et Numéris. Il s'agit d'un livre vraiment indispensable aux utilisateurs

et programmeurs de réseaux, aussi bien qu'à ceux qui débutent en ce domaine. Vendu 298 F, son prix n'est pas surévalué.

Andrew Tannenbaum est professeur à l'Université libre d'Amsterdam. Il a publié plusieurs ouvrages qui font maintenant autorité (*Architecture de l'ordinateur, les Systèmes d'exploitation, Conception et mise en œuvre, Minix, guide de référence*), plusieurs articles chez ACM et IEEE (dont « Distributed Operating Systems » et « A Practical Tool Kit for Making Portable Compilers », et a écrit un système d'exploitation en C ainsi qu'un compilateur portable sous plusieurs machines (PC, Atari, Mac...), qui sont vendus avec leur code source. C'est donc l'ouvrage d'un véritable spécialiste que traduit aujourd'hui Interéditions.

Service lecteur n° 113

PC ET COMPATIBLES, LE BIOS MIS A NU

Sous un titre accrocheur et une déclaration d'intention revendicatrice se cache un livre dont on ne sait que penser. Michel Cottini y a aussi bien réuni des renseignements précieux (le plan du BIOS d'IBM) que des informations erronées (les variables du BIOS en mémoire vive sont confondues avec celles du DOS, dont aucun livre ne donne la signification ni l'emplacement exact en mémoire). En outre, alors que l'on s'attendait à un ouvrage centré uniquement sur le BIOS, on trouve également des généralités sur le DOS, le Basic, l'architecture interne de l'IBM-PC. Fait de bric et de broc, ce livre de la nouvelle collection Sybex contient un peu de tout, et rien de complet. A consulter éventuellement pour le plan du BIOS.

Service lecteur n° 114

PETITS FORMATS A PETITS PRIX

Les éditions PSI lancent trois nouvelles collections de poches : les « Guides rapides », « Initiation facile », et « Micro-mémento » à la couverture gris sobre. Elles présentent en bref les grands systèmes d'exploitations, langages, et applicatifs (SQL, Unix, MS-DOS, Works, Turbo-Pascal, Word, et même Gw-Basic, etc.). Destinés à un public de débutants ou de gens pressés, ces ouvrages permettent d'avoir sous la main les principaux renseignements nécessaires au fonctionnement de base du logiciel. Ces guides de poche sont vendus entre 25 et 50 F et comptent de 100 à 200 pages. Les deux premiers « Guides rapides » sont consacrés à Lotus 1-2-3 version 2.2 (d'Anne-Céline Guiheneuf, 200 pages, 55 F), et Framework III (de Daniel-Jean David, 130 pages, 50 F).

Service lecteur n° 115

DEMARRER AVEC WORD POUR WINDOWS (VERSIONS 2 ET 3.0)

A peine Word pour Windows et Windows version 3.0 étaient-ils lancés sur le marché, que Michael Boom publiait ce livre aux éditions Microsoft Press (Penguin Books). Jean Guillemin l'a aussitôt traduit pour PSI, et nous disposons maintenant du guide (quasi) officiel de Word pour Windows. Le moins que l'on puisse dire, c'est qu'il s'agit d'un livre richement illustré (environ une copie d'écran toutes les deux pages), et fourni en exemples (nombre de lettres « professionnelles », avec force marges, styles, formats). L'ouvrage s'avère exhaustif : de la configuration nécessaire (un 286 avec 640 Ko de mémoire, 10 Mo de disponible sur le disque dur et une souris... optionnelle !) au maniement des macros. Certains aspects avancés sont cependant négligés (impression de formules mathématiques par exemple). Il s'agit d'un bon livre, pas trop cher (295 F), mais réservé aux utilisateurs débutants (ayant « perdu » les manuels ?). Une

VIDEO SHOP
GROUPE ALLIANCE

Une rentrée exceptionnelle

Toute une gamme de PC
à des prix
défiant toute concurrence

PC 20 III — **7 990 F TTC**
+ Lecteur 3"1/2 + Lecteur 5"1/4 + Disque dur 20 Mo
+ Moniteur monochrome HERCULES

PC 30 III — **12 990 F TTC**
+ Disque dur 20 Mo + Moniteur couleur VGA couleur 14"
+ Imprimante 24 aiguilles STAR LC 24/10

PC 50 SX AT 386 SX — **15 990 F TTC**
+ Disque dur 40 Mo + Moniteur Couleur VGA 14"

Imprimante DESKJET Plus — **6 990 F TTC**

MAINTENANCE
Notre service dépannage pourra satisfaire tous vos besoins en matière de maintenance informatique sous 8 heures.
Un service Hot Line est également à votre écoute 24h/24
Appelez le **43.27.27.88**

3615 VS

RESEAUX
Tout l'univers des réseaux et de la communication en informatique maîtrisée par une équipe de spécialistes
Réseaux NOVELL, ETHERNET, 3 COM, TOKEN RING
Devis gratuit sur demande

Tous nos prix s'entendent TTC - GARANTIE 2 ANS
REPRISE DE VOTRE ANCIEN MATERIEL - **CREDIT - LEASING - FORMATION**
Catalogue contre 3 timbres à 2,20 F

POUR COMMANDER
Par téléphone : 42.86.03.44
Par Minitel : 3615 Code VS
Par courrier : Vidéo-Shop
VPC BP 105 75749 Paris cedex 15

Dernière minute !
Compatible AT 286 Dual AT 201
5 990 F TTC

4 MAGASINS A VOTRE SERVICE !!!
AU CENTRE : 47 / 50, rue de Richelieu - 75001 PARIS - M° Palais-Royal
AU SUD : 251, boulevard Raspail - 75014 PARIS - Métro Raspail
A L'OUEST : 7, rue de l'église - 92200 NEUILLY - M° Pont-de-Neuilly
A L'EST : 260, rue de Charenton - 75012 PARIS - M° Daumesnil

MAGASINS OUVERTS DU LUNDI AU SAMEDI
de 9 heures à 20 heures sans interruption

(1) 42.86.03.44
FAX 42.86.01.22

Cerclez info pub 15

bonne surprise : la couverture souple est enfin digne de celles des éditions originales, et la mise en page est particulièrement réussie.

Service lecteur n° 116

AIDE-MEMOIRE WORKS 2

L'ouvrage consacré à Works 2 est représentatif des titres de la collection Aide-mémoire de Cedic/PSI Pratique à utiliser, il contient l'essentiel des informations relatives au logiciel traité. Le format allongé permet de présenter un maximum d'informations sur une même page.

Sur le plan de l'organisation générale, l'auteur, Jean-Louis Marx, a consacré un chapitre à chacun des modules qui composent le logiciel intégré Works 2. Les principales commandes du traitement de textes, du tableur, et de la base données sont présentées clairement, et agrémentées de quelques schémas. Le reste des 48 pages de l'ouvrages traitent des caractéristiques générales de Works 2, des fonctions de communication, et des implications entre les divers modules (publipostage, intégration de feuilles de calculs ou de graphiques dans un fichier texte, etc.). Au bout du compte, un ouvrage utile à un prix raisonnable (55 F).

Service lecteur n° 117

LIVRES INTEL

Il y a encore peu de temps, les ouvrages de référence des microprocesseurs Intel n'étaient disponibles qu'auprès du fabricant américain. C'est désormais terminé, puisqu'Intel a confié la distribution de ses livres aux libraires spécialisées. Le Monde en Tique, notamment, les vend tous, et tient son stock à jour. Parmi ces livres, on retiendra surtout les Hardware Reference Manuals ainsi que les Programmer's Reference Manuals pour les microprocesseurs 80286, i386, i486, et i860. L'Iapx 88 Book fait en revanche figure de parent pauvre face aux livres d'autres éditeurs (Sybex, par exemple).

Ces livres ne sont malheureusement pas traduits en français, et coûtent environ 250 F chaque. Mais le jeu en vaut la chandelle, et peu de programmeurs sauraient se passer de tels ouvrages.

Service lecteur n° 118

LIAISONS MAC-PC

L'échange de données entre le monde PC et Macintosh constitue un véritable parcours du combattant. Cet ouvrage de 128 pages (par Ph. Escoffier-Gentille et J.N. Clot, éditions Armand Colin/*Info PC*, 85 F) tente de faire le point sur les différentes solutions proposées à l'utilisateur. La première partie traite du transfert d'informations à proprement parler. Les auteurs y décrivent la communication directe par câble ou *via* un modem, les lecteurs externes, les réseaux locaux hétérogènes. Une fois le transfert d'une machine sur l'autre effectué, il reste à réaliser le transcodage de données. C'est donc l'objet de la seconde partie où sont présentés différents logiciels de transcodage. Ce petit manuel permettra ainsi un réel gain de temps à tous ceux qui souhaitent rapprocher Apple et Big Blue.

Service lecteur n° 119

GESTION DU PERSONNEL, MODELES DE CONTRATS

La présence de cet ouvrage dans le cadre de cette rubrique peut sembler inopportune, puisqu'il ne traite pas directement d'informatique. Le seul lien objectif avec cette discipline étant constitué par la disquette d'accompagnement.

Néanmoins, ce titre, issu de la collection Applications de gestion éditée par Sybex, trouvera place dans les services administratifs des sociétés, informatiques ou non. Il s'agit d'un ouvrage pratique de droit du travail, présentant succinctement et clairement les droits et les obligations des deux parties, employés et employeurs.

Tous les domaines sont abordés : conventions collectives, inspection du travail, droit d'expression, droits syndicaux, intéressement aux profits, etc.

Une large part des quatre cents pages du livre est consacrée à la rédaction des contrats de travail. Les auteurs abordent bien entendu les différents types de contrats — durée déterminée ou indéterminée, apprentissage, contrats de formation, saisonniers, etc. — et les clauses indispensables : période d'essai, durée du travail, rémunération, etc. Le livre s'achève avec le chapitre relatif aux ruptures de contrat, licenciements ou démissions. Une disquette d'accompagnement contient les formulaires et les lettres types permettant de gérer efficacement le personnel d'une entreprise.

Le prix un peu élevé de cet ouvrage, 478 F, ne saurait constituer un obstacle pour les acquéreurs potentiels que sont les sociétés.

Service lecteur n° 120

INITIATION ACTIVE A DBASE III PLUS ET DBASE IV

Cet ouvrage de M. Guignard aux éditions PSI (269 pages, 245 F) s'adresse aussi bien à l'utilisateur sans connaissances informatiques particulières qu'à celui possédant déjà un certain bagage. A l'aide de la disquette d'exemples fournie avec le livre, la prise en main de dBASE s'effectue de façon rapide et concrète au cours de la première partie. La seconde s'applique à décrire la conception d'une application. La troisième est un manuel de référence (ensemble des commandes et fonctions) simplifié. L'auteur présente les différents niveaux d'utilisation de dBase : menus, commandes, programmes, dans les deux versions du logiciel. Les méthodes à employer avec dBase III Plus et dBase IV sont décrites l'une après l'autre, ce qui permettra à l'utilisateur connaissant dBase III Plus une évolution en douceur vers dBase IV. Le langage SQL n'est, quant à lui, pas abordé. La lecture de ce livre paraît donc utile à tous.

Service lecteur n° 121

AIDE-MEMOIRE DBASE IV, SYSTEME ET PROGRAMMATION

En format allongé — pas forcément idéal pour le rangement — et en 79 pages (collection Aide-mémoire), ce petit manuel fournira de nombreux conseils à tous ceux qui souhaitent développer une application dBase. Pas un mot néanmoins sur le langage SQL. Ce mémento de S.D. Camey, aux éditions Cedic/Nathan, constitue un complément pratique au volumineux manuel de référence de dBase IV.

Service lecteur n° 122

SQL : LE LANGAGE STRUCTURÉ D'INTERROGATION

Les auteurs, C.J. Hursch et J.L. Hursch, s'attachent à décrire le langage SQL. L'ouvrage aux éditions Masson (176 pages, 160 F) intéressera non seulement ceux qui pratiquent SQL en tant qu'utilisateur, mais aussi les informaticiens qui conçoivent et développent des applications complètes à l'aide de ce langage. Les fondements théoriques mathématiques et logiques sont expliqués, ainsi que l'utilisation concrète des commandes. Les différents composants de SQL sont successivement abordés : le langage d'interrogation de données, le langage de définition

de données, le langage de manipulation de données, le langage de contrôle de données. Les commandes sont illustrées à l'aide d'exemples. Le sujet est traité avec beaucoup de sérieux et de rigueur. L'achat du livre constitue de ce fait une excellente acquisition. Peut-être une approche un peu universitaire.

Service lecteur n° 123

UNIX : SAVOIR FAIRE

Les ouvrages de présentation d'Unix sont toujours les bienvenus lorsque l'on a été confronté à la documentation originale de ce système d'exploitation ; deux bons mètres d'étagère, aucune volonté didactique, le tout en anglais. Ce livre de Richard Stoeckel, aux éditions Armand Colin (344 pages, 189 F), permet une excellente prise en main du système. Après une présentation générale, l'auteur parvient à dresser un panorama des outils (éditeurs de textes, filtres, etc.), des commandes et du langage de programmation (programmes shell appelés également scripts) fournis par Unix. De nombreux exemples facilitent la compréhension du propos. Excellent ouvrage donc, la référence absolue sur le sujet restant cependant l'environnement de programmation Unix par Brian Kerninghan et Rob Pike chez Interéditions.

Service lecteur n° 124

LE GUIDE MICRO APPLICATION : WORD 5

La présence d'un guide peut rassurer l'utilisateur d'un logiciel aussi performant et complexe que Word 5. A condition toutefois qu'il se révèle plus synthétique et plus clair que les manuels originaux. Ce Guide, écrit par Helmut et Manfred Tornsdorf, répond tout à fait à ce critère de qualité.

Basé sur une approche progressive des possibilités du logiciel, il constitue un manuel de référence tout à fait acceptable. Si l'on peut lui reprocher de manquer d'illustrations (où sont passées les copies d'écran et les schémas légendés ?), un système d'icônes permet de marquer les points particuliers, erreurs à éviter, points délicats, conseils pratiques. Une large part de l'ouvrage est occupée par un rappel des commandes de Word 5, une vue des différents menus et sous-menus, ainsi qu'un récapitulatif des raccourcis au clavier. En plus de sa vocation de manuel de référence, ce livre peut donc avantageusement jouer le rôle d'un mémento. Ce guide relativement complet (près de 400 pages), et proposé à un prix attractif (128 F), ne présente pas les habituels défauts des ouvrages traduits. La formulation reste sobre et compréhensible.

Service lecteur n° 125

■ « Le Grand Livre de DR-DOS », aux éditions Micro Application, dresse, en 528 pages, un panorama du système d'exploitation de Digital Research, à l'exclusion des fonctions introduites par la version 5.0 (de Jörg Schieb, 245 F). ***Service lecteur n° 126***

■ Les 346 pages de « PageMaker PC par la pratique » ne sont pas de trop pour saisir les subtilités de la mise en page assistée par ordinateur (d'Antonia Stacy Jolles, aux éditions Sybex, 278 F). ***Service lecteur n° 127***

■ « Fonctions avancées sur Word 4 et 5 » introduit les fonctionnalités de pointe du traitement de textes de Microsoft (de Nelly Cadeddu et Philippe Vaugouin, aux éditions Eyrolles, 280 pages, 250 F). ***Service lecteur n° 128***

■ « Microsoft Works 2 PC Micro-référence » est un aide-mémoire qui rassemble l'essentiel des commandes et fonctions de Works 2 en 50 pages (de Henri Chêne, aux éditions Sybex, 68 F). ***Service lecteur n° 129***

■ « Le Livre des compatibles AT-286 » ou comment tout connaître des 286, race prestigieuse condamnée à une fin prochaine ? (de Jörg Schieb, aux éditions Micro Application, 546 pages, 195 F). ***Service lecteur n° 130***

■ L'informatique au service de l'art ? « Art et Ordinateur » d'Abraham A. Moles dresse un panorama des implications entre les diverses formes d'expression artistique (graphisme, films d'animation, architecture, musique) et la technologie informatique (aux éditions Blusson, 320 pages, 145 F). ***Service lecteur n° 131***

■ « La Programmation sous Windows » plonge au cœur du kit de développement de Windows, fort de plus de 400 fonctions. Un ouvrage de Michel Desmadril, paru aux éditions Eyrolles (328 pages, 315 F). ***Service lecteur n° 132***

VIDEO SHOP

Bonne Rentrée

Une rentrée encore plus forte sur toute la gamme des PC 1512 et 2386

3615 VS

Achetez en septembre Payez en décembre

Extension mémoire PC 1512 490 F TTC

AMSTRAD — PROMO VACANCES

PC 2086 + Disque dur 30 M° + logiciel Works + windows + souris avec moniteur mono 12 MD 8 990 F
avec moniteur couleur 14 HRCD 10 990 F

Gratuit imprimante DMP 3250 DI

PC 2286 AT 286, 1 M° de mémoire, Disque Dur 40 M° + logiciel Works avec moniteur VGA monochrome 12" 11 990 F
avec moniteur VGA couleur 14" HR 13 990 F

Gratuit imprimante 24 aiguilles LQ 3500

PC 2386 AT 286 + Works, 4 M° de mémoire vive, Disque Dur 65 M° + Moniteur VGA couleur 14" 24 900 F

PC 1512 DD couleur + Imprimante DMP 3160 7 490 F
PC 1640 HD couleur + Imprimante DMP 3160 10 990 F

Toute la nouvelle gamme PC 3000 en démonstration permanente

UNIQUE !!! FAX à partir de 3 990 F HT contacter Mme Bianpain (1) 46.40.73.27

PC 2386 HD 4 Mo + disque dur 65 Mo + Windows + Works
Monochrome 22 900 F
Couleur 24 900 F

FABULEUX IMPRIMANTE LASER • Canon LBP4 9 990 F TTC

Tous nos prix s'entendent TTC - GARANTIE 2 ANS
REPRISE DE VOTRE ANCIEN MATERIEL - CREDIT - LEASING - FORMATION
Catalogue contre 3 timbres à 2,20 F

POUR COMMANDER
Par téléphone : 42.86.03.44
Par Minitel : 3615 Code VS
Par courrier : Vidéo-Shop
VPC BP 105 75749 Paris cedex 15

Tablette graphique A4 2 990 F
Lecteur 5 1/4 ext 360 Ko 990 F
Lecteur 5 1/4 ext 1,2 Mo 1 290 F
Disquettes 3 1/2 DF DD 4,90 F l'unité

4 MAGASINS A VOTRE SERVICE !!!

AU CENTRE : 47 / 50, rue de Richelieu - 75001 PARIS - M° Palais-Royal
AU SUD : 251, boulevard Raspail - 75014 PARIS - Métro Raspail
A L'OUEST : 7, rue de l'église - 92200 NEUILLY - M° Pont-de-Neuilly
A L'EST : 260, rue de Charenton - 75012 PARIS - M° Daumesnil

MAGASINS OUVERTS DU LUNDI AU SAMEDI de 9 heures à 20 heures sans interruption

(1) 42.86.03.44
FAX 42.86.01.22

Cerclez info pub 16

le système d'exploitation OS/2

LE VRAI MULTITACHE

Le multitâche est certainement un des plus importants apports de OS/2. Pourtant c'est un des aspect les plus oubliés. Cet article décrit le fonctionnement et les avantages des possibilités multitâches de OS/2.

Quel est l'apport d'un système multitâche comme OS/2 par rapport à un système monotâche comme MS-DOS ? La réponse est simple : la possibilité d'exécuter simultanément plusieurs programmes ou parties de programme augmente l'efficacité globale du système. Les temps morts des programmes (réaction de l'utilisateur, accès disque, etc.) sont utilisés par les autres programmes.

un système d'exploitation multitâche est capable d'exécuter simultanément plusieurs programmes.

Par exemple, pendant qu'une tâche gère le dialogue avec l'utilisateur et reçoit la liste des commandes, une seconde tâche peut exécuter les travaux correspondants aux commandes. L'utilisateur pourra lancer une nouvelle commande sans attendre que la première soit terminée. De plus, souvent, le temps mis à taper la commande et ses paramètres est supérieur au temps d'exécution proprement dit. Avec deux tâches (ou plus), l'aspect purement séquentiel est rompu pour devenir partiellement parallèle. Le bénéfice pour l'utilisateur se traduit par un meilleur temps de réponse du système.

Un autre exemple classique d'application du fonctionnement multitâche est la réalisation en arrière plan d'un travail long : impression, analyse statistique, réindexation, tri...

Surtout, un système multitâche simplifie énormément la tâche du programmeur lors de la réalisation de systèmes complexes et dynamiques. Par exemple, l'écriture d'un programme d'émulation de terminal devient évident : une tâche s'occupe de recevoir des caractères en provenance de la ligne de communication et de les placer à l'écran pendant qu'une seconde tâche scrute le clavier et transmet à la ligne de communication les caractères introduits par l'utilisateur. Plus besoin de se préoccuper du cas où un caractère provient de la ligne de communication pendant que l'on teste le clavier.

Il est assez facile d'implémenter des architectures de programme du type client / serveur avec un système multitâche. Le programmeur écrit une seule tâche qui effectue le travail requis par les clients. Cette tâche est exécutée en parallèle autant de fois qu'il y a de clients simultanés.

Il ne faut pas confondre systèmes multitâche et multi-utilisateur. Les seconds permettent à plusieurs personnes physiques de lancer des programmes et les utilitaires du système au départ de différents postes de travail. Un système multi-utilisateur est forcément multitâche : il y a au minimum une tâche par utilisateur. Par contre, un système multi-utilisateur ne dialogue qu'avec un seul utilisateur *via* la console du système.

Il est clair qu'un système multitâche permet d'écrire des programmes qui, eux, sont multi-utilisateurs. Dans ce contexte, un programme est lancé plusieurs fois au départ de la console, chaque exemplaire s'occupant d'un utilisateur connecté au système par l'intermédiaire, par exemple, d'un port de communication asynchrone.

Pour simplifier, disons que dans un système multi-utilisateur, tout programme peut être exécuté de n'importe quel poste de travail, sans devoir prévoir quelque chose de spécial. Alors que pour obtenir le même résultat dans un système multitâche, le programmeur doit tenir compte de la possibilité pour le programme de travailler non pas avec l'écran et le clavier ordinaire (la console), mais avec un périphérique spécial connecté *via* une interface.

OS/2 comme système multitâche augmente l'efficacité du matériel

En fin de compte, un système d'exploitation multi—utilisateur digne de ce nom doit offrir des facilités qui permettent d'interdire à certains postes l'accès aux données et aux programmes des autres. La sécurité est fondamentale, un système de permissions est nécessaire.

OS/2 est sans conteste un système multitâche, mono-utilisateur. Toutes les fonctions d'entrée / sortie et d'écran / clavier sont prévues pour travailler avec l'écran et le clavier du PC. Il n'est pas possible — ou difficilement —d'utiliser ces fonctions avec un terminal connecté au système.

OS/2 comporte deux niveaux de fonctionnement multitâche : les « processus » et les « threads ». Il est évident que OS/2 permet d'exécuter simultanément plusieurs programmes (ou processus), chacun étant constitué d'au moins un thread.

Pour bien comprendre les différences entre thread, processus et programme, il est nécessaire de faire un pas en arrière et de réviser les possibilités des processeurs qui animent les machines OS/2 : 80286 et i386. Pour l'heure OS/2, fait fonctionner les i386 et i486 comme un super 80286, sans profiter de leur architecture de 32 bits (voir la présentation dans ce numéro d'OS/2 2.0).

OS/2 comporte deux niveaux de multitâche : les threads et les processus

Le 80286 peut fonctionner dans deux modes : le mode d'adressage réel et le mode protégé. Le premier est bien connu puisque MS-DOS fonctionne dans ce mode (adressage limité à 1 Mo). Le second est celui qui nous intéresse, puisque OS/2 ne travaille qu'avec lui.

Dans le mode protégé, le 80286 est capable d'adresser 16 Mo de mémoire décomposée en segments de 16 à 65536 octets. Une adresse est composée d'un sélecteur de segments et d'un offset (position à l'intérieur du segment). Le sélecteur de segments n'est rien d'autre qu'un index dans une table de descripteurs de segments. Cette table contient, entre autres, la taille, l'adresse de départ, et des permissions d'accès pour chaque segment.

Pour les instructions exécutées par le processeur, les adresses ne correspondent donc pas réellement à la mémoire physique. Il existe une étape de translation (réalisée automatiquement par la partie MMU du 80286) qui transforme une adresse logique (celle du programme) en une adresse physique (celle du matériel).

Si on change le contenu de la table de translation — les adresses logiques restant identiques — le même programme accédera à une toute autre partie de la mémoire physique.

Revenons maintenant aux processus : OS/2 attribue à chacun d'entre-eux une table différente, dont les segments sont communs (mémoire partagée) ou non avec les autres processus.

Un processus ne peut en aucun cas accéder à une partie (physique) de la mémoire centrale sans avoir, dans sa table de translation, une entrée qui pointe sur cette partie. Il y a isolation complète des espaces d'adressages des processus exécutés par OS/2.

Si à ce mécanisme on ajoute celui des niveaux d'exécution (RING) on comprend que la protection est totale. Le 80286 dispose de 4 niveaux d'exécution ayant chacun des permissions plus ou moins grandes. Par exemple, il faut être au niveau 0 (tout y est permis) pour pouvoir modifier les tables de translation d'adresses. OS/2 fait fonctionner les processus au niveau 3 (le plus restrictif). Il est impossible à un processus de modifier les tables de translation d'adresses et donc d'accéder à de la mémoire qui ne lui aurait pas été allouée par le système d'exploitation.

Pour terminer le rapide tour du contexte d'exécution d'un processus, il est encore nécessaire d'introduire la notion de console virtuelle. OS/2 est prévu pour faire fonctionner simultanément plusieurs programmes, par exemple une comptabilité et un traitement de textes. Pour l'utilisateur, il est intéressant de bien séparer les écrans de ces deux applications. Aussi OS/2 maintient-il un écran virtuel (une session) pour chaque application (seize écrans au maximum).

OS/2 gère consoles virtuelles.

Lorsqu'une application écrit à l'écran, elle le fait dans son écran virtuel (session, screen group). Le système OS/2 permet à l'utilisateur de changer d'écran virtuel et donc de programme, par simple appui sur une touche.

Il faut aussi savoir que lorsque Presentation Manager (PM) est installé, il fonctionne dans une seule session en mode graphique. Toutes les applications PM travaillent dans cette session unique. C'est PM qui gère l'affichage des sorties de chaque application dans une ou l'autre fenêtre graphique.

Un programme est un ensemble d'instructions (le code) et de données dont une image est stockée sur disque, dans un fichier. Lorsque ce programme est chargé en mémoire et que OS/2 a tout préparé pour lan-

cer son exécution, le programme devient un processus.

Les instructions qui constituent un processus sont exécutées en séquence, avec les ruptures de séquences imposées par les branchements conditionnels ou non, et les appels de sous-programmes. On peut assimiler un programme en fonctionnement à un flux d'instructions (thread) en cours d'exécution. C'est parfait, mais limité. OS/2 permet à un processus de démarrer l'exécution simultanée (à l'échelle humaine) de plusieurs flux d'instructions.

En clair, cela signifie qu'un processus en cours d'exécution peut dire à OS/2 (faire un appel système) de démarrer l'exécution d'un autre flux d'instructions (un thread).

La différence fondamentale entre un processus et un thread se situe au niveau de l'adressage : deux processus sont complètement indépendants en matière d'espace d'adressage alors que tous les threads d'un processus partagent le même espace d'adressage. En pratique, les variables globales d'un processus sont accessibles par n'importe quel thread dudit processus et par aucun autre thread d'aucun autre processus.

La seule chose qui soit privée à un thread au sein d'un processus est sa copie des registres du processeur, sa pile et son état d'exécution.

Chaque processus dispose de son propre espace d'adressage. Tous les threads d'un même processus partagent le même espace d'adressage

Le noyau de OS/2 se charge de toutes les protections nécessaires entre processus, mais chaque thread d'un processus est seul responsable et ne doit pas perturber le fonctionnement des autres threads du même processus.

Voyons à présent l'interface (API) offerte par OS/2 au programmeur en vue de la création, du fonctionnement, de la gestion des processus et des threads.

La description exhaustive de tous les appels systèmes sort largement du cadre de cet article. Pour donner un ordre d'idées au lecteur, disons simplement que l'essentiel de la description des appels système de OS/2 tient dans les trois volumes du manuel: « Microsoft Operating System/2 Programmer's Reference », comportant respectivement 740, 564 et 430 pages de petits caractères, et adoptant une présentation très compacte.

Nous nous en tiendrons donc dans cet article à une description de principe des principaux appels système relatifs à l'aspect multitâche d'OS/2.

Au même titre que n'importe quel autre processus, l'interpréteur de commandes ou la fenêtre « Start Programs » de Presentation Manager exécute l'appel système « DosExecPgm » pour lancer l'exécution d'un nouveau processus (un programme).

```
DosExecPgm(
  PCHAR pchFailName,          /* où placer un éventuel message     */
  SHORT cbFailName            /* taille du tampon pour le message */
  USHORT fExecFlags,          /* mode de fonctionnement            */
  PSZ    pszArgs,             /* ligne de commande (arguments)     */
  PRESULTCODES prescResults, /* où placer les codes de retour     */
  PSZ    pszPgmName)          /* nom du programme                  */
```

```
VOID DosExit(
   USHORT fTerminate,      /* indique si termine un ou tous     */
   USHORT usExitCode)      /* donne le code de retour au père */
```

« DosExecPgm » demande à OS/2 de charger en mémoire un fichier contenant un programme (fichier .EXE), d'initialiser le contexte de fonctionnement (registres, pile, données) et de lancer l'exécution au point d'entrée du thread principal. L'exécution peut se faire de diverses manières : asynchrone, synchrone, en arrière plan, etc. C'est l'argument fExecFlags qui détermine le mode de fonctionnement.

Un processus s'exécute de manière asynchrone par rapport à celui qui l'a créé si, et seulement si, les deux processus s'exécutent simultanément. A contrario, un processus synchrone arrête l'exécution du processus père pendant son exécution (comme sous MS-DOS).

Un processus en exécution en arrière plan est complètement détaché de son processus père et de sa console virtuelle (screen group). Si le processus père meure, le fils continue d'exister indépendamment. Ce mode de fonctionnement est utilisé en général pour créer des « démons », comme par exemple les spooler (queue impression).

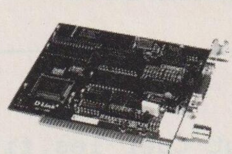
Chaque processus reçoit un numéro arbitraire qui permet de l'identifier. On parle du PID. Ce numéro est placé par « DosExecPgm » dans un des membres de la structure « prescResults ».

Pour terminer l'exécution d'un processus, il faut terminer l'exécution de chacun des threads qui compose le processus. Un appel système permet à un thread de se terminer ou au contraire de terminer l'exécution de tous les threads du processus, donc terminer le processus lui-même. Si un thread est une fonction, un simple « return » termine le thread en question.

Un thread ou un processus qui se termine rend à son père (celui

Nouveau produit KOMELEC

CARTES RESEAUX



Carte ETHERNET 10 mbp	
— 16 bits	2 660 F
— 8 bits	2 420 F
Carte ARCNET	1 235 F

KOMELEC

La liaison informatique

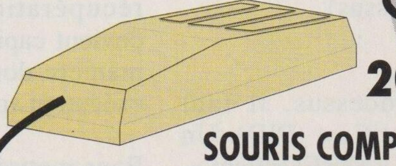
4, rue Yves TOUDIC
75010 Paris
du lundi au samedi
de 10 h à 12 h 30
et de 13 h 30 à 19 h
Métro République

Tél. : (1) 42 08 63 10
(1) 42 08 54 07 +
Fax : (1) 42 08 59 05

Atelier de câblage
Atelier de câblage
Atelier de câblage
Atelier de câblage
Atelier de câblage

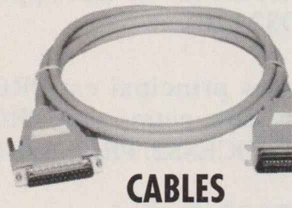
A VOTRE DISPOSITION

Conception de tous types de câbles
Conception de tous types de câbles
Conception de tous types de câbles
Conception de tous types de câbles
Conception de tous types de câbles



200 F

SOURIS COMPATIBLE MICROSOFT



CABLES

CABLE IMPRIMANTE PC	Prix U.
1,80 m	48,00 F
3,00 m	80,00 F
5,00 m	120,00 F
7,00 m	130,00 F
CABLE SÉRIE 25 M/M OU MF	
1,80 m	48,00 F
3,00 m	80,00 F
5,00 m	120,00 F
7,00 m	130,00 F
10,00 m	222,00 F

Convertisseur SP ou PS	644 F
Printer Buffer 256 K 2E/1S	2127 F
Printer Buffer 64 K 1 PC/1 IMP	1093 F
Mini Buffer EB110 64 K	750 F
Mini Buffer EB410 256 K	1526 F

CONNECTEURS ET CAPOTS

	Prix U.
DB09 MALE/FEMELLE	4,00 F
DB15 "	5,00 F
DB 15 HD "	10,00 F
DB19 "	8,00 F
DB23 "	8,00 F
DB25 "	6,00 F
DB37 "	12,00 F
DB50 "	25,00 F
CAPOT DB 9	6,60 F
CAPOT DB 15	6,70 F
CAPOT DB 25	6,80 F

CHANGEURS DE GENRE (MINI)

CG 25 F/25 F	45 F
CG 25 M/25 M	"
CG 9 F/9 F	"
CG 9 M/9 M	"

Adaptateur de modem AT 09 F/25 M	45 F
Adaptateur de modem AT 09 M/25 F	45 F
Mini-testeur RS 232 7 lignes	105 F
Adaptateur moniteur VGA 9 F/15 HDM	65 F
Cordon minitel PC et PS 2	95 F

MEMOIRES

41256-10	35 F	27256	30 F
41256-12	32 F	27C256	32 F
43256-8	145 F	27512	66 F
43256-10	135 F	41 1000	100 F

DATA SWITCH AUTOMATIQUE SANS BUFFER



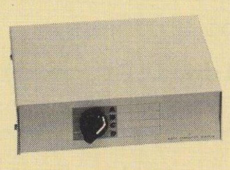
8E/1S Série	1290 F
4E/1S Série	1095 F
8E/1S Parallèle	1190 F
4E/1S Parallèle	960 F

(prévoir un câble par micro)

DATA SWITCH AUTOMATIQUE AVEC BUFFER 256 K

4E/1S	Parallèle	1928 F
8E/1S	Parallèle	2385 F
4E/2S	Parallèle	2275 F
8E/2S	Parallèle	2650 F

DATA SWITCH MANUEL



4E/1S
350 F

PRIX PAR QUANTITE - NOUS CONSULTER

Minimum de commande : 100 F - Administrations : sur bon de commande - Catalogue (précisez PC ou ATARI) 30 F remboursé à partir de 200 F d'achat. Port : + 40 F jusqu'à 3 kg. Prix indicatifs.
Extrait de notre catalogue connectique : les prix s'entendent TTC.

One - O - One - 46 04 88 78

Cerclez info pub 24

```
USHORT DosKillProcess(
       USHORT fScope,            /* indique s'il faut tuer les fils */
       PID    pidProcess)        /* identifiant du processus à tuer */
```

quil'avait démarré) un code de retour. Si le père est l'interpréteur de commandes, ce code est accessible via la variable ERRORLEVEL.

Mais un processus ne se termine pas forcément de lui-même, il est parfois nécessaire de le « tuer » au départ d'un autre processus. (voir encadré ci-dessus)

Pour tuer un processus, il faut connaître son identifiant (PID). Un processus reçoit de « DosExecPgm », les identifiants des processus fils qu'il lance. Lorsque l'on tue un processus, on peut vouloir le tuer — disons— personnellement, ou bien avec toute sa descendance. La nuance est de taille puisqu'un processus peut en démarrer d'autres, qui chacun à leur tour en démarrent encore, créant ainsi une véritable arborescence de processus.

Chaque processus qui se termine rend à son père un code de retour. C'est l'appel système « DosCwait » qui permet de récupérer ce code à toutes fins utiles. (voir encadré ci-dessous)

« DosCwait » permet soit d'attendre qu'un processus fils se termine et de récupérer son code de retour, soit de récupérer le code de retour du dernier processus fils qui s'est terminé, s'il y en a un.

Il est fréquent qu'un processus père doive se synchroniser sur la fin de l'exécution d'un processus fils. Si le père a besoin des résultats produits, par le fils, il faut bien qu'il en attende la fin d'exécution.
Plus souvent encore, le code de retour d'un processus indique si tout s'est bien passé comme prévu. La récupération du code de retour devient capitale dans le choix de la manière dont un père poursuit son exécution après avoir lancé un fils.

Pour mettre en avant ces quelques appels système, voyons un exemple élémentaire et démonstratif : un programme démarre l'exécution de plusieurs fils. Les fils ont pour mission d'imprimer quelques messages à intervalles réguliers. Tant que le père ne détecte pas la fin de l'exécution des fils, il imprime lui aussi un message à intervalles réguliers. Lorsque tous les fils ont imprimé chacun leurs messages, le père découvre que tous ses fils sont morts, il l'indique par un dernier message avant de se terminer lui-même.

Nous verrons par la suite le même programme, mais organisé en threads au lieu de processus.

Lors de l'exécution, apparaissent sur l'écran les divers messages imprimés par les différents processus. Les délais programmés sont suffisamment longs pour voir la succession des messages.
Le programme comprend trois processus, dont deux sont identiques, mis à part le contenu des messages. Chaque processus correspond à un fichier exécutable obtenu par compilation des sources et édition des liens entre le module objet généré par le compilateur et les librairies correspondant aux fonctions du compilateur (LLIBCE.LIB) et aux appels système (OS2.LIB).

Le processus principal est PROCESS0, les deux autres sont PROCESS1 et PROCESS2. PROCESS1

et PROCESS2 sont obtenus par compilation du même source — PROCESS1.C — par le truchement d'un symbole défini à la ligne de commande et donnant le numéro du processus. Remarquez l'emploi dans le source, de la possibilité de concaténation des chaînes de caractères simplement juxtaposées.

Les processus font ici appel à OS/2 pour toutes les opérations d'écriture des messages. On aurait pu employer la fonction standard printf(), ce qui aurait eu pour effet d'allonger sensiblement la taille de l'exécutable et aurait privé le programme de la démonstration supplémentaire de la fonction « DosWrite() » qui permet d'écrire dans un fichier ou à la console

```
USHORT DosCwait(
 USHORT fScope,                  /* combien de processus à attendre    */
 USHORT fWait,                   /* indique s'il faut attendre         */
 PRESULTCODES prescResults,      /* où placer les résultats            */
 PPID   ppidProcess,             /* où placer le PID du terminé        */
 PID    pidWaitProcess)          /* PID du processus à attendre        */
```

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
Programme:       PROCESS0.C
Auteur:          F. Piette
Création:        29/04/90
Objet:           Démonstration des processus sous OS/2.
                 Fonctionne conjointement avec PROCESS1.C
Compilation:     Microsoft C V5.1
                 CL /c /AL /W2 /G2 /Zp PROCESS0.C
                 LINK /NOI /NOD PROCESS0.OBJ,,NUL,LLIBCE.LIB OS2.LIB;

 /* *  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
#define INCL_DOS
#include <stdio.h>
#include <string.h>
#include <os2.h>

/* Définit une constante donnant le nombre de processus à lancer          */
#define PNB     2
/* Liste des fichiers contenant les processus                              */
PSZ     pszProcessName[PNB] = { "PROCESS1.EXE", "PROCESS2.EXE" };
/* Liste des arguments passés aux processus                                */
PSZ     pszProcessArgs[PNB] = { "PROCESS1 \0",  "PROCESS2 \0"  };

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Inversion d'une chaîne de caractères ("abcd" devient "dcba")            */
CHAR *reverse(CHAR s[])
{
    CHAR  c;
    SHORT i, j;

    for (i = 0, j = strlen(s) - 1; i < j; i++, j—) {
        c    = s[i];
        s[i] = s[j];
        s[j] = c;
    }
    return(s);
}

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Conversion d'un nombre entier en une chaîne de caractères               */
CHAR *itoa(SHORT sVal, CHAR achBuf[])
{
    SHORT i, sign;

    /* Enregistre le signe du nombre, garde la valeur absolue              */
    if ((sign = sVal) < 0)
        sVal = -sVal;

/* Construction chiffre par chiffre du nombre                                    */
    i = 0;
    do {
    achBuf[i++] = (CHAR)(sVal % 10 + '0');
    } while ((sVal /= 10) > 0);

    /* Si le nombre était négatif, ajoute le signe moins                   */
    if (sign < 0)
        achBuf[i++] = '-';

    /* Termine la chaîne par le zéro habituel                              */
    achBuf[i] = '\0';

    /* Retourne la chaîne obtenue                                          */
    return(reverse(achBuf));
}

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Impression à la console d'un message composé de trois parties:          */
/* un texte, un nombre entier puis un second texte.                        */
void PrintMsg(PSZ pszPart1, SHORT sVal, PSZ pszPart2)
{
    CHAR    achBuffer[128];              /* Pour contenir le message       */
    CHAR    achBuf[32];                  /* Pour convertir le nombre       */
    USHORT  cbWritten;                   /* Requis par OS/2                */

    strcpy(achBuffer, pszPart1);
    strcat(achBuffer, " ");
```

```
    strcat(achBuffer, itoa(sVal, achBuf));
    strcat(achBuffer, " ");
    strcat(achBuffer, pszPart2);
    strcat(achBuffer, "\r\n");
    DosWrite(1, achBuffer, strlen(achBuffer), &cbWritten);
  }

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
void main(void)
{
    /* Deux messages utiles                                                   */
    PSZ         pszMsg1 = "0: Processus principal démarré\r\n";
    USHORT      cbMsg1  = strlen(pszMsg1);
    PSZ         pszMsg2 = "0: Fin du processus principal\r\n";
    USHORT      cbMsg2  = strlen(pszMsg2);

    USHORT      cbWritten;              /* Compteur de bytes écrits          */
    CHAR        achFailName[128];       /* Message d'erreur                   */
    RESULTCODES rescResults;            /* Résultats d'exécution              */
    USHORT      usCount;                /* Compteur                           */
    USHORT      usRunning;              /* Compteur de processus lancés       */
    PID         pidProcess;             /* PID du processus terminé           */
    USHORT      usErrorCode;            /* Code d'erreur rendu par OS/2       */

    /* Imprime le message d'introduction                                      */
    DosWrite(1, pszMsg1, cbMsg1, &cbWritten);
    /* Lance l'exécution des processus fils                                   */
    for (usCount = usRunning = 0; usCount < PNB; usCount++) {
        /* Indique le lancement                                               */
        PrintMsg("0: Lancement du processus fils", usCount + 1, "");

        /* Lance effectivement le processus fils                              */
        usErrorCode = DosExecPgm(achFailName,
                                 sizeof(achFailName),
                                 EXEC_ASYNCRESULT,
                                 pszProcessArgs[usCount],
                                 0,
                                 &rescResults,
                                 pszProcessName[usCount]);

        /* En cas d'erreur, imprime le numéro de code rendu par OS/2          */
        if (usErrorCode) {
            PrintMsg("0: Erreur au lancement du processus",
                     usCount + 1,
                     pszProcessName[usCount]);
            PrintMsg("0: Code OS/2", usErrorCode, "");
        }
        else
            usRunning++;                /* Comptabilise le processus          */

        /* Imprime le message donnant le PID du processus fils                */
        PrintMsg("0: PID du processus fils =",
                 rescResults.codeTerminate,
                 "");
    } /* Imprime les messages du processus principal, délai de 1/2 seconde     */
    /* jusqu'à ce que tous les processus fils soient morts.                   */
    usCount = 1;
    do {
        PrintMsg("0: Message", usCount, "du processus principal");
        DosSleep(500L);
        usCount++;

        /* Boucle de test des processus fils qui se terminent. Tant que       */
        /* l'on a détecté la mort d'un fils, on reteste tout de suite         */
        /* pour voir si un autre n'est pas mort. Sinon on imprime             */
        /* un message puis on fait un délai avant de retester.                */
        for (;;) {
            /* Demande le code de retour d'un processus, sans attente,        */
            /* ce qui donne une erreur s'il n'y a pas de processus mort       */
            usErrorCode = DosCwait(DCWA_PROCESS, DCWW_NOWAIT,
                                   &rescResults, &pidProcess, 0);

        /* Si aucun processus n'est mort, un code d'erreur est rendu          */
            if (usErrorCode != 0)       /* Alors on arrête de tester          */
                break;
```

3615 FREEWARE



* sur 3615 Freeware, on trouve même la Bible en Anglais à télécharger !! (sur PC)

DES CENTAINES DE LOGICIELS POUR VOTRE MICRO

FREEWARE Vous permet de télécharger sur votre micro des centaines de logiciels avec le plus grand confort : transfert fiable et rapide, près de 6 K-octets par minute (pour moins de 25 francs, vous téléchargez un fabuleux jeu d'échecs !). Grâce au 3615 FREEWARE constituez vous une bibliothèque de logiciels de qualité pour votre micro IBM PC ou compatibles, et ATARI ST.

LE LIEU DE RENCONTRE DES PASSIONNES DE MICRO

Retrouvez vous dans la messagerie et les forums en direct, les rubriques vous permettant de rentrer en contact avec d'autres passionnés de micro, les boites aux lettres et les jeux.

GAGNEZ UN ATARI ST

Jouez sur le 3615 FREEWARE au jeu du Pacman et du Snake et gagnez chaque mois un micro ordinateur ATARI ST.
Et si vous avez un minitel 2, vous serez surpris par la qualité du graphisme !

L'INDISPENSABLE KIT DE TELECHARGEMENT QUICKER

Pour télécharger tous ces logiciels qui vous attendent, vous devez posséder le kit de téléchargement QUICKER. Ce kit est constitué d'une disquette contenant le programme QUICKER et d'un câble de liaison micro-minitel. Pour commander ce kit, remplissez ci-dessous le bon de commande.
Si vous possédez déjà le cable, branchez-vous sur le 3615 FREEWARE pour commander la disquette uniquement.

AVANT - PREMIERE :

En avant-première tous les mois, téléchargez les démonstrations et découvrez les prochains jeux de vos éditeurs préférés.

Bon de commande à découper et à retourner sous enveloppe affranchie à : CPIO 3615 Freeware, 39 rue St Antoine 75004 PARIS.

☐ OUI, je désire recevoir le kit complet, (câble, disquette Quicker, documentation plus un logiciel en cadeau) au prix spécial de 149 F *

Je possède déjà un câble micro minitel et je souhaite uniquement recevoir le logiciel Quicker pour mon micro :

29 F * ☐ Disquette PC ou Compatible 5 pouce 1/4

NOM .. 34 F * ☐ Disquette PC ou Compatible 3 pouce 1/2

PRENOM .. 34 F * ☐ Disquette ATARI ST

Adresse .. Etage Bâtiment

Code Postal Ville .. Tél. (facultatif)

* Joindre un chèque à l'ordre de CPIO.

Cerclez info pub 18

• C.P. n° 56918

```
        /* Pas d'erreur, un processus est donc bien mort            */
        PrintMsg("0: Processus de PID =",
                 pidProcess,
                 "terminé");
        }
        /* Décompte le processus qui vient de mourir                */
        usRunning—;
      } while (usRunning > 0);

   /* Imprime le message de fin et rend la main à OS/2              */
    DosWrite(1, pszMsg2, cbMsg2, &cbWritten);
    DosExit(EXIT_PROCESS, 0);
}

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
Programme:      PROCESS1.C
Auteur:         F. Piette
Création:       29/04/90
Objet:          Démonstration des processus sous OS/2.
                Fonctionne conjointement avec PROCESS0.C
Compilation:    Microsoft C V5.1 (ANSI-C)
                Pour générer PROCESS1.EXE:
                CL /c /D PROCESS=\"1\" /AL /W2 /G2 /Zp PROCESS1.C
                LINK /NOI /NOD PROCESS1.OBJ,PROCESS1.EXE,NUL,LLIBCE.LIB OS2.LIB;
                Pour générer PROCESS2.EXE:
                CL /c /D PROCESS=\"2\" /AL /W2 /G2 /Zp PROCESS1.C
                LINK /NOI /NOD PROCESS1.OBJ,PROCESS2.EXE,NUL,LLIBCE.LIB OS2.LIB;

                Pour générer PROCESS2.EXE:
                CL /c /D PROCESS=\"2\" /AL /W2 /G2 /Zp PROCESS1.C
                LINK /NOI /NOD PROCESS1.OBJ,PROCESS2.EXE,NUL,LLIBCE.LIB OS2.LIB;
```

Listing du programme PROCESS0.C

Pour détecter la fin de tous les processus fils lancés, le processus principal les compte lors de leur lancement (variable usRunning). Puis, dans la boucle d'impression des messages, il fait un appel à « DosCwait() » qui indique une erreur si aucun processus n'est mort. Lorsque tous les fils sont morts, la boucle d'impression des messages se termine et le processus principal prend fin.

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
       #define INCL_DOS
       #include <stdio.h>
       #include <string.h>
       #include <os2.h>

       /* Si la ligne de commande ne précise par de numéro de processus..*/
       #ifndef PROCESS
       #   define PROCESS "1"            /* ...on prend le numéro          1*/
       #endif
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
void main(void)

    USHORT  cbMsg1  = strlen(pszMsg1);
   PSZ     pszMsg1 = PROCESS ": Processus " PROCESS " démarré\r\n";
    USHORT  cbMsg1  = strlen(pszMsg1);

    PSZ     pszMsg2 = PROCESS ": Message du processus " PROCESS "\r\n";
    USHORT  cbMsg2  = strlen(pszMsg2);
    PSZ     pszMsg3 = PROCESS ": Fin du processus " PROCESS "\r\n";
    USHORT  cbMsg3  = strlen(pszMsg3);
    USHORT  cbWritten;
    USHORT  usCount;

    DosWrite(1, pszMsg1, cbMsg1, &cbWritten);

    for (usCount = 0; usCount < 4; usCount++) {
        DosWrite(1, pszMsg2, cbMsg2, &cbWritten);
        DosSleep(1000L);
    }
       DosWrite(1, pszMsg3, cbMsg3, &cbWritten);
    DosExit(EXIT_PROCESS, 0);
}
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

Listing du programme PROCESS1.C

```
USHORT DosCreateThread
(PFNTHREAD pfnFunction(VOID,          /* Pointeur sur la fonction  */
PTID       ptidThread,                /* Où rendre l'identificateur */
PBYTE      pbThrdStack;               /* Pointeur sur la pile       */
```

L'avantage —- qui peut parfois être un inconvénient —- de procéder ainsi est la sécurité. Chaque processus possède son propre espace d'adressage. Impossible donc de corrompre les données d'un autre processus ou d'OS/2 lui-même.

Les principaux défauts des processus sont les mauvais temps de réponse, la place occupée sur le disque et en mémoire, et l'éclatement en de nombreux exécutables.

Le mauvais temps de réponse que l'on pourrait définir comme le temps qui s'écoule entre l'instruction qui requiert le lancement d'un processus et le moment où le processus exécute sa première instruction, provient du fait que chaque processus se trouve dans un fichier exécutable différent et que OS/2 doit charger en mémoire ce fichier avant de pouvoir démarrer l'exécution proprement dite.

La place occupée sur disque et en mémoire peut devenir importante si on ne met pas en oeuvre les outils sophistiqués offerts par OS/2 : librairie à chargement dynamique (DLL) et code partagé. Sans cela, chaque processus contient une copie des librairies de fonctions et des routines communes.

L'argument ptidThread est utilisé par OS/2 pour rendre l'identificteur (le numéro) du thread créé.
L'exécution du thread commence à la première instruction de la fonction dont on donne l'adresse comme premier argument.

Pour qu'un thread puisse s'exécuter indépendamment des autres, il a besoin d'une pile. Elle sert à stocker les arguments et l'adresse de retour des fonctions appelées par le thread, ainsi que pour y stocker des variables locales (stack frame).

Dernier inconvénient, toujours lié aux fichiers : chaque processus se trouvant dans un fichier séparé, une application finit par être dispersée. Pourtant, dans certains cas, cela devient un avantage en permettant de remplacer facilement un processus par un autre, plus à jour ou plus efficace. De même, on peut livrer un programme par module sans aucune difficulté.

L'utilisation des threads comble totalement ces lacunes. Un thread peut être considéré comme une tâche qui fonctionne en parallèle avec les autres. Au niveau source du programme, chaque thread est en fait une fonction. Plusieurs threads peuvent du reste exécuter la même fonction simultanément, à condition que son code soit réentrant (de même que celui de toutes les fonctions ou procédures qu'il appelle).

Un thread est une simple fonction qui s'exécute en parallèle avec le reste du programme

Les threads qui constituent un processus sont tous dans le même exécutable, à moins qu'ils ne soient dans une librairie à chargement dynamique (DLL). Ils partagent tous le même espace d'adressage, et donc les mêmes variables globales et objets alloués par OS/2 (fichiers, pipes, sémaphores, etc.).
La fonction « DosCreateThread » permet de lancer l'exécution d'un thread. Sa syntaxe est la suivante (voir encadré ci-dessus).
La pile d'un thread doit avoir une taille suffisante pour les variables locales et les appels de fonction. Elle doit même avoir une taille de 2 Ko supérieure au minimum requis par le thread, ceci pour permettre le fonctionnement de OS/2 lorsqu'il prend la main.

Un thread se termine soit parce que la fonction qui le constitue se termine (instruction return), soit parce que « DosExit » a été appelée. Nous avons déjà vu cette fonction pour les processus. C'est la même qui est employée, mais son premier argument doit être EXIT_THREAD au lieu de EXIT_PROCESS.

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

Programme:      THREAD.C
Auteur:         F. Piette
Création:       29/04/90
Objet:          Démonstration des threads sous OS/2.
Compilation:    Microsoft C V5.1
                CL /c /ALw /W2 /G2 /Zp THREAD.C
                LINK /NOI /NOD THREAD.OBJ,,NUL,LLIBCE.LIB OS2.LIB;

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

```
#define INCL_DOS
#define INCL_SUB
#include <stdio.h>
#include <string.h>
#include <os2.h>

#pragma check_stack (off)

/* Definit une constante donnant le nombre de threads à lancer        */
#define TNB      2

#define STACK_SIZE        2560

USHORT      usRunning;                  /* Compteur de threads lancés      */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Inversion d'une chaîne de caractères ("abcd" devient "dcba")   */
CHAR *reverse(CHAR s[])
{
      CHAR  c;
      SHORT i, j;

      for (i = 0, j = strlen(s) - 1; i < j; i++, j—) {
            c    = s[i];
            s[i] = s[j];
            s[j] = c;
      }
      return(s);
}
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Conversion d'un nombre entier en une chaîne de caractères          */
CHAR *itoa(SHORT sVal, CHAR achBuf[])
{
      SHORT i, sign;

      /* Enregistre le signe du nombre, garde la valeur absolue        */
      if ((sign = sVal) < 0)
            sVal = -sVal;

    /* Construction chiffre par chiffre du nombre                     */
      i = 0;
      do {
            achBuf[i++] = (CHAR)(sVal % 10 + '0');
      } while ((sVal /= 10) > 0);

      /* Si le nombre était négatif, ajoute le signe moins            */
      if (sign < 0)
            achBuf[i++] = '-';
  /* Termine la chaîne par le zéro habituel                          */
      achBuf[i] = '\0';

      /* Retourne la chaîne obtenue                                   */
      return(reverse(achBuf));
}
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Impression à la console d'un message composé de trois parties:     */
/* un texte, un nombre entier, puis un second texte.                  */
void PrintMsg(PSZ pszPart1, SHORT sVal, PSZ pszPart2)
{
      CHAR     achBuffer[128];           /* Pour contenir le message     */
      CHAR     achBuf[32];               /* Pour convertir le nombre     */
      USHORT   cbWritten;                /* Requis par OS/2              */
strcpy(achBuffer, pszPart1);
      strcat(achBuffer, " ");
      strcat(achBuffer, itoa(sVal, achBuf));
      strcat(achBuffer, " ");
      strcat(achBuffer, pszPart2);
strcat(achBuffer, "\r\n");
      DosWrite(1, achBuffer, strlen(achBuffer), &cbWritten);
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
void Thread(void)

      USHORT  usCount;
      USHORT  usMyThreadID;
      PIDINFO pidi;

      DosGetPID(&pidi);
```

LA PUISSANCE À VOTRE PORTÉE

MAINE INFORMATIQUE

MI 286/12 MHZ
1 Mo de RAM
DD 40 Mo
Carte vidéo Hercules
Ecran paper white
Souris + Tapis
GEM 3.0 + FIRST WORD PLUS
DOS 5.0
8 900 F HT
VGA couleur
Carte vidéo 16 Bits
Ecran VGA 14 pouces
11 900 F HT

MI 286/16 MHZ
1 Mo de RAM
DD 40 Mo
Carte vidéo Hercules
Ecran paper white
Souris + Tapis
GEM 3.0 + FIRST WORD PLUS
DOS 5.0
9 300 F HT
VGA couleur
Carte vidéo 16 Bits
Ecran VGA 14 pouces
12 500 F HT

MI 386/SX16 MHZ
1 Mo de RAM
DD 40 Mo
Carte vidéo Hercules
Ecran paper white
Souris + Tapis
WINDOW 3.0
DOS 5.0
11 500 F HT
VGA couleur
Carte vidéo 16 Bits
Ecran VGA 14 pouces
13 900 F HT

MI 386/20 MHZ
4 Mo de RAM
DD 40 Mo
Carte vidéo VGA
VGA monochrome
Souris + Tapis
WINDOW 3.0
DOS 5.0
19 900 F HT
VGA couleur
Carte vidéo 16 Bits
Ecran VGA 14 pouces
21 900 F HT

MI 386/25 MHZ
4 Mo de RAM
D D 40 Mo
Carte vidéo VGA 512 K
Ecran VGA monochrome
Souris + Tapis
1 Floppy 1,2 Mo + 1 Floppy 1,44
DOS 5.0 + WINDOW 3.0
29 900 F HT
VGA couleur
Carte vidéo 16 Bits 512k
Ecran VGA NEC 3D /14 pouces
32 900 F HT

Controleurs disque dur / floppy. Clavier 102 touches AZERTY, support coprocesseur. Assemblés et testés aux Etats Unis. Mini-Tower ou Tower sans supplément !

Toutes nos configurations, sont livrées avec deux ports série un port parallele, support coprocesseur 287 pour 286 et 387 pour 386. (EXTENSIBLE EN MEMOIRE JUSQU'A 8 Mo, SLOTS D'EXTENSIONS) LIM 4.0 EMS, un floppy 1,2 Mo 5p 1 /4 ou 3p1/2, clavier AZERTY 102 touches DOS, GEM et Souris installés, Documentation et manuel en Français.

EXTENSIONS MEMOIRES

CARTE EXTENSION MEMOIRE POUR PC/AT (16 BITS)

BOCA RAM AT PLUS

Livrée équipée de 2 méga de mémoire extensible à 8 méga , MS LIM 3.2 et 4.0, (Driver fourni), fonctionne avec tous les logiciels (Lotus 1-2-3 Windows, Desqview). Support Dos et Os/2. Etend la mémoire conventionnelle de 512 k à 640 k. Driver pour Ramdisk et Mémoire tampon imprimante.

Carte équipée de deux méga	2450 F HT
Le méga supplémentaire	850 F HT

Existe aussi pour XT (8 bits)

Carte pour PS2 BUS MCA

Carte équipée de 2 Mégas pour 50/60	2950 F HT
Carte équipée de 2 Mégas pour 50Z	2950 F HT
Carte série parallèle BUS MCA	1850 F HT

BARETTES EXTENSION 1 Mo (SIM ou SIP)

Tous types PC/AT (100 Ns à 70 Ns)	690 F HT
Barettes extensions 256 K	285 F HT
CHIPS MEMOIRE	
256 Ko (9X256 Ko)	198 F HT
1 Méga (9X1 Méga)	650 F HT

CARTES MÉMOIRES

IBM (8530/8555/8570/8580)
COMPACQ (tous modèles)
TOSHIBA (tous modèles)

IMPRIMANTES

IMPRIMANTE LASER CANON LBP4
4 Pages minute, Polices Vectorielles — **7990 F HT**

IMPRIMANTE HP LASER JET SERIE II P
Equipée de 2 Mégas — **11 390 F HT**

CARTES D'EXTENSIONS POUR IMPRIMANTES LASER

CANON LBP 8 II , II R et II T
HP LASER JET SERIE II, IID, IIP, et III

CARTE MEMOIRE EQUIPEE DE 2 MEGA	2950 F HT
CARTE MEMOIRE EQUIPEE DE 4 MEGA	4750 F HT

EXTENSIONS MÉMOIRE
IBM 4019 et APPLE LASER WRITER
PRIX PROMOTIONNELS

Digital Research
WE MAKE COMPUTERS WORK

DR DOS 5.0

DOS NOUVELLE GENERATION

✓ View Max interface graphique utilisateur à base d'icônes à la norme CUA vous permet de lancer un programme en cliquant sur l'icône

✓ Memory Max gestion de mémoire optimisée laissant 620 K à l'utilisateur et driver chargé en mémoire haute, y compris les logiciels de gestion de réseau

NOVELL NETWARE chargé avec DR DOS : les applications nécessitant une mémoire importante comme XEROX VENTURA PUBLISHER OU AUTOCAD, peuvent être executées sans problèmes sous NOVELL

✓ DISQUE CACHE permet d'améliorer de façon significative le temps d'accès au disque dur

✓ FILE LINK vous permet a l'aide d'un simple câble série de transférer des fichiers a 115 200 bauds

✓ Partition DOS jusqu'à 512 méga

✓ L.I.M. 4.0 expanded memory compatible avec tous systèmes i 386, i 486 et PS 2 80286

✓ EDITOR éditeur pleine page. Créer et modifier ▪ BAT, ▪ SYS ou un fichier est une chose aisée

✓ PASSWORD vous permet de protéger vos fichiers et directory

✓ MEM affiche votre table mémoire

990 F HT

Aptec Technology®

Base station 286 — **9990 F HT**
286/12 Mhz 1 Mo - DD 40 Mo
ECRANS VGA couleur 14 pouces

PC VII 386/33 — **33390 F HT**
386/33 Mhz 4 Mo - DD 110 Mo
ECRANS VGA couleur 14 pouces

LA SECURITE

... Garantie 1 an sur site
pièces et mains d'œuvre
(sur toute la France par TELCI SA)

COPROCESSEURS

- Coprocesseurs totalement compatibles pin à pin avec INTEL
- 2/4 plus rapides
- Architecture haute performance sur 80 bits
- Très faible consommation
- Autonomie prolongée pour les micro-ordinateurs portables grace à une fonction de veille

COPROCESSEURS DE I.I. TECHNOLOGY

coprocesseur arithmétique 2 fois plus rapide

2C87-8 MHZ	2C87-10 MHZ	2C87-12 MHZ	2C 87-20 MHZ
1590 HT	1790 HT	2100 F HT	2 450 F HT

FASMATH 83D87 de Cyrix

coprocesseur arithmétique actuellement le plus rapide (4fois)

SX16 MHZ	DX20 MHZ	DX25 MHZ	DX33 MHZ
2150 F HT	2750 F HT	3450 F HT	4250 F HT

MONITEUR

Toutes nos configurations couleurs sont équipées de **moniteur V.G.A.** de marque **OCEANIC 14 Pouces (36 cm) Pitch 0.31** Made in France.

Le Moniteur Oceanic	**2 990 F HT**
Moniteur NEC 3D	**4 990 F HT**

PROMOTION

DISQUE DUR 40 Mo 28ms	2 400 F HT
CARTE VGA 16 bits 256 k extensible à 512 k	990 F HT
SCANNER A MAIN SCAN MAN LOGITECH	1250 F HT

... Livraison 5 à 10 Jours

MAINE INFORMATIQUE
118, Ave du maine 75014 PARIS
Tél. : 42 79 94 27/Fax. : 42 79 94 09

TOULOUSE
Place du Capitol 35, rue du Taur - 31000 TOULOUSE
Tél. : 62 27 04 37

Réalisation POINT COMPO Tél : 43 20 44 61

Cerclez info pub 19

Toutes les marques citées sont des marques déposées®.

```
 DosGetPID(&pidi);
    usMyThreadID = pidi.tid;

PrintMsg("Thread", usMyThreadID, "démarré");

for (usCount = 0; usCount < 4; usCount++) {
PrintMsg("Message du thread", usMyThreadID, "");
        DosSleep(1000L);

PrintMsg("Fin du thread", usMyThreadID, "");

    DosEnterCritSec();
    usRunning—;
    DosExitCritSec();

    DosExit(EXIT_THREAD, 0);
}
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
void main(void)
{
    USHORT      cbWritten;              /* Compteur de bytes écrits       */
    USHORT      usCount;                /* Compteur                       */
    USHORT      usErrorCode;            /* Code d'erreur rendu par OS/2   */
    USHORT      usMyThreadID;
    USHORT      usThreadID;
    SEL         sel;

    usMyThreadID = 1;

   /* Imprime le message d'introduction                                   */
PrintMsg("Thread", usMyThreadID, "démarré");

    /* Lance l'exécution des autres threads                                */
    for (usCount = usRunning = 0; usCount < TNB; usCount++) {
        /* Indique le lancement                                            */
        PrintMsg("Thread 1: Lancement du thread", usCount + 2, "");

    /* Alloue de la mémoire pour la pile                                   */
        usErrorCode = DosAllocSeg(STACK_SIZE, &sel, SEG_NONSHARED);

/* En cas d'erreur, imprime le numéro de code rendu par OS/2               */
 if (usErrorCode) {
PrintMsg("Thread 1: Erreur d'allocation mémoire",
                     usErrorCode,
                     "");
                DosExit(EXIT_PROCESS, 0);
                }
/* Lance effectivement le processus fils                                   */
        usErrorCode = DosCreateThread(Thread,
                              &usThreadID,
                              MAKEP(sel, STACK_SIZE));

 /* En cas d'erreur, imprime le numéro de code rendu par OS/2              */
if (usErrorCode) {
 PrintMsg("Thread 1: Erreur au lancement du Thread",
                           usCount + 2,
PrintMsg("Thread 1: Code OS/2", usErrorCode, "");

}
else
 usRunning++;                   /* Comptabilise le processus               */

/* Imprime le message donnant le PID du processus fils                     */
      usThreadID,"")

  /* Imprime les messages du processus principal, délai de 1/2 seconde     */
    /* jusqu'à ce que tous les threads soient morts.                       */
    do {
        PrintMsg("Message du Thread", usMyThreadID, "");
        DosSleep(500L);
    } while (usRunning > 0);

    /* Imprime le message de fin et rend la main à OS/2                    */
    PrintMsg("Tread 1: Fin du thread", usMyThreadID, "");
    DosExit(EXIT_THREAD, 0);
}
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

Listing du programme THREAD.C

La difficulté majeure rencontrée dans l'écriture d'un programme multithread provient des routines de la librairie standard. Dans la version 5.10 du compilateur Microsoft C, seule une petite partie des routines sont réentrantes. Microsoft fournit une librairie baptisée LLIBCMT.LIB qui contient les versions réentrantes des routines de la librairie standard, notamment « printf() ». Le programme de démonstration THREAD.C présenté dans cet article n'utilise aucune routine de la librairie C standard non réentrante. En particulier, les messages sont imprimés directement par un appel système d'OS/2. A l'heure de l'écriture de ces lignes, Microsoft a présenté la version 6.0 de son compilateur pour laquelle les librairies ont été retravaillées. Il nous a donc paru plus intéressant de nous consacrer aux fonctions d'OS/2 en matière de threads, plutôt qu'aux fonctions spécifiques au Microsoft C v5, appelées à évoluer. La pile nécessaire au fonctionnement des threads est obtenue en demandant à OS/2 d'allouer un segment mémoire de la taille appropriée. c'est l'appel « DosAllocSeg() ». Tous les threads lancés exécutent la même fonction simultanément. C'est la fonction « Thread() » qui est réentrante (pas de variable globale). Le comptage des threads lors de leur lancement (variable usRunning) permet de déterminer lorsqu'ils sont tous morts. Contrairement à la version « processus » du programme (voir ci-dessus), chaque thread décrémente lui-même la variable usRunning, elle-même testée par le thread principal (thread˘1). Là se trouve le piège de la programmation multitâche ! Pour décrémenter une variable, il faut la lire, soustraire 1, puis la réécrire. Or il se peut qu'OS/2 arrête l'exécution juste après qu'un thread ait lu la variable et avant qu'il ne l'ait modifiée et que le thread qui prend la main fasse exactement la même chose. Dans ce cas de figure, la variable n'est décrémentée qu'une seule fois. Pour éviter ce piège, il suffit d'arrêter momentanément le basculement de tâches. Cela se fait avec l'appel « DosEnterCritSec() ». Dès que cet appel est exécuté, et jusqu'à ce que l'appel « DosExitCriSec() » soit réalisé, OS/2 laisse la main au thread qui fonctionne à ce moment. Impossible donc qu'un autre prenne la main au mauvais moment.

Ce mécanisme fonctionne parfaitement et reste efficace si le travail pendant la période critique est très bref. Ce qui est le cas dans notre démonstration. Si, par contre, le travail est plus long, il faut utiliser un autre mécanisme : les sémaphores. Mais ceci est une autre histoire. Ce rapide survol permet comprendre la complexité d'OS/2, notamment par rapport à MS-DOS, mais aussi d'apprécier ses bienfaits et possibilités d'optimisation du système que ce système d'exploitation apporte.

François Piette

N.A.I.T. Informatique
4, Avenue DAMIETTE - 95110 SANNOIS
Tel: 47. 98. 76. 50 / 34. 10. 97. 97 - Fax: 34. 10. 68. 16

PROMOTION
(Dans la limite de nos stocks)

TANDON PCA 12 SL 20 MO MONOCHROME
+ IMPRIMANTE CITIZEN SWIFT 24
+ LOGICIEL CIEL COMPTA-GESTION
+ WINDOWS 286 AVEC SOURIS
+ 2 JOURNEES DE FORMATION
= 13.960,00 F.HT

TANDON PAC 286/12 VGA COULEUR
+ IMPRIMANTE CITIZEN SWIFT 24
+ FRAMEWORK EXECUTIVE
+ WINDOWS 286 AVEC SOURIS
+ 2 JOURNEES DE FORMATION
+ 1 DISQUE DUR AMOVIBLE 30 MO
+ 1 LECTEUR 3" 1/2 OU 5" 1/4
= 24.995,00 F.HT

Cerclez info pub 20

Spécial OS/2

PROGRAMMATION SOUS OS/2 PRESENTATION MANAGER

Cet article s'attache à décrire, au travers d'un exemple pratique, la programmation sous OS/2 Presentation Manager. Le jeu de Tic-Tac-Toe servira à examiner le fonctionnement de Presentation Manager.

Dans le jeu de Tic-Tac-Toe, deux joueurs utilisent un damier de neuf cases et placent tour à tour et respectivement une croix et un cercle. Celui qui arrive à aligner trois de ces symboles remporte la partie. C'est un jeu très simple qui va nous permettre de nous concentrer sur la programmation sous Presentation Manager, plutôt que sur une stratégie de jeu.

L'objet du programme est d'ouvrir une fenêtre, d'y tracer un damier et de permettre aux deux joueurs de placer à tour de rôle une croix ou un cercle. La souris permet de désigner la case où placer le symbole. Un menu permet de terminer ou de recommencer une partie. Au clavier, les touches de déplacement du curseur permettent aussi de positionner les symboles, la touche ENTER d'en enregistrer un.

La touche ALT donne accès au menu. Ensuite, la touche de l'alphabet présentée en majuscule dans le menu permet de déclencher l'action



Ecran typique de OS/2 Présentation Manager. On y voit la fenêtre " Start Programs " qui permet de lancer des applications, la fenêtre " Task Manager " qui montre toutes les applications qui tournent dans la machine, la fenêtre du jeu de Tic-Tac-Toe pris en exemple dans cet article, une fenêtre d'une ligne de l'interpréteur de commandes, et les icônes des applications qui tournent en arrière - plan. Notez qu'il y a deux jeux de Tic-Tac-Toe en fontionnement : un actif à l'écran et l'autre dormant, minimisé sous forme d'icône.

désignée (cette procédure est standard sous Presentation Manager et Windows).
Pour être complet, le programme comprend une boîte « A propos », les commandes complètes de déplacement et de taille de la fenêtre, ainsi que sa minimisation ou maximisation. Bien entendu, les graphiques sont en couleur.

qu'est-ce que Presentation Manager?

Comme chacun le sait, Presentation Manager est la partie interface utilisateur graphique d'OS/2. Cette extension est partie intégrante d'OS/2 depuis la version 1.1, parue fin 1988.
Presentation Manager est une réminiscence d'un programme de recherche développé chez Xerox dans leur centre de recherches de Palo Alto (PARC) depuis les années 70 jusqu'au début des années 80. Apple et Microsoft ont largement contribué à développer la renommée des interfaces graphiques « à fenêtres » avec, respectivement, le Macintosh et Windows.
L'interface utilisateur de Presentation Manager reste très proche de celle de Windows 2.0, Windows 286, Windows 386 et maintenant Windows 3.0. L'utilisateur final sera donc immédiatement productif lors du passage du système d'exploitation MS-DOS à OS/2.
Pour le développeur de programmes, Presentation Manager propose une panoplie complète d'appels système (environ 500 API) qui permettent de créer des fenêtres et de les manipuler. La philosophie de programmation est largement empruntée — pour ne pas dire dérivée — de celle de Windows. Le programmeur habitué à Windows sera donc très rapidement efficace sous Presentation Manager.
Outre le fenêtrage, Presentation Manager offre des facilités considérables dans le domaine du graphisme avec le GPI (Graphics Programming Interface). GPI est un système très sophistiqué adapté du GDDM (Graphics Data Display Manager) et du GCP d'IBM (3270 Graphics Control Program). Ici, ce sont les programmeurs des gros systèmes IBM qui se retrouveront dans un environnement familier.
Bien sûr, comme Presentation Manager tourne sous OS/2, il profite de facto des possibilités multitâches, de la mémoire virtuelle, de la mémoire physique extensible à 16 Mo, et des mécanismes de communications interprocessus d'OS/2. Ainsi, un même programme développé sous MS-DOS / Windows 3.0 et sous OS/2 / Presentation Manager sera plus efficace, plus rapide et plus interactif dans le second cas. Cependant, et c'est très important, l'interface utilisateur (le mode d'emploi, en quelque sorte) sera le même dans les deux cas.
Pour terminer ce rapide tour d'horizon, précisons encore que l'interface utilisateur et les API de Presentation Manager font partie intégrante de l'ambitieuse SAA d'IBM (System Application Architecture). Si la société IBM gagne son pari — et il n'existe pas de raison qu'elle n'y parvienne pas —, alors l'interface utilisateur de Presentation Manager se retrouvera sur toute la gamme IBM, du PC au mainframe, en passant par les mini-ordinateurs.
N'oublions pas non plus de mentionner que le C est le langage principal de développement sous Presentation Manager. Cela n'empêche pas de travailler en Basic ou en Pascal, mais Microsoft livre tout ce qui est nécessaire pour travailler en C, ce qui n'est pas le cas avec les autres langages.

La philosophie de programmation

Dans une application conventionnelle, le développeur écrit son logiciel de telle manière qu'il pilote l'utilisateur au gré des besoins du programme. Lorsque des données doivent être introduites, le programme appelle une fonction qui interroge l'utilisateur et attend sa réponse. En fonction de celle-ci, le programme continue son travail d'une manière ou d'une autre, procédant ainsi par questions/réponses du début à la fin.
Dans un programme sous Presentation Manager, c'est l'utilisateur qui pilote le programme, et non l'inverse. La fonction principale du logiciel consiste à attendre une action de l'utilisateur. Cette dernière est notifiée au programme par l'envoi d'un message à l'attention de la fenêtre concernée.
Une action de l'utilisateur peut être une frappe au clavier, un déplacement de la souris, une pression sur un des boutons de la souris, la sélection d'un menu ou d'un élément d'un menu, et bien d'autres choses encore.
Pour présenter le même concept d'une autre manière, donnons un exemple : dans un programme conventionnel, lorsqu'il faut lire un caractère tapé au clavier par l'utilisateur, il suffit d'appeler une fonction « getchar() » en C qui bloquera le système jusqu'à ce que l'utilisateur tape un caractère. Sous Presentation Manager, le programme continue de se dérouler normalement — donc par une simple boucle d'attente d'un message — jusqu'à ce que l'utilisateur daigne appuyer sur une touche au clavier. A ce moment, un message indiquant quelle touche a été enfoncée est envoyé au programme. Il ne reste plus qu'à traiter ce message, et déclencher l'action qui convient.
Sous Presentation Manager, chaque objet connu du système et manipulé par un programme est soit prédéfini, soit créé explicitement par le programme. Pour faire référence à un objet, le système et le programmeur utilisent un identificateur (« handle » en anglais) qui est généralement un nombre entier, parfois un pointeur. Lorsque l'objet n'est plus nécessaire, il est détruit.
Ce mécanisme est analogue à celui employé de longue date pour les fichiers : pour utiliser un fichier, le programmeur doit appeler une fonc-

tion (« fopen ») qui reçoit, entre autres paramètres, le nom du fichier et son unité disque, et qui retourne un identificateur. Chaque accès ultérieur au fichier s'effectue en ne précisant plus le nom du fichier, mais bien l'identificateur rendu par le système. Lorsque l'usage du fichier est terminé, le programmeur « ferme » le fichier en appelant une fonction (close) avec l'identificateur. Ce dernier est rendu libre et sera éventuellement employé à nouveau par le système pour faire référence à un autre fichier.
Sous Presentation Manager, l'analogie est parfaite avec les fenêtres : pour employer une fenêtre, le programmeur doit la créer en appelant la fonction « WinCreateStdWindow() » en lui donnant une série d'arguments qui la caractérisent. En retour, Presentation Manager rend un identificateur qui sera utilisé partout où une référence à la fenêtre doit être opérée. Après usage, la fenêtre est détruite en appelant la fonction « WinDestroyWindow() ». L'identificateur devient invalide et sera éventuellement réutilisé par la suite par Presentation Manager pour identifier une autre fenêtre.
Le programmeur devra associer à chaque fenêtre une procédure de traitement des messages qui aura pour objet d'effectuer la tâche associée à la fenêtre.
Une même procédure peut servir pour autant de fenêtres que nécessaire. En pratique, l'association se fait en deux temps : d'abord enregistrer une ou plusieurs « classes » de fenêtres (WinRegisterClass), puis au moment de la création des fenêtres (WinCreateStdWindow), un des arguments indique pour la fenêtre créée à quelle classe elle appartient et donc quelle est la procédure de traitement des messages.
La répartition des messages aux fenêtres se fait par Presentation Manager au moyen de la fonction « WinDispatchMsg ». Le coeur du programme de toute application Presentation Manager est donc une boucle de récupération de messages (WinGetMsg) et de répartition de ces messages (WinDispatchMsg) dans les fenêtres qui y correspondent.
Chaque procédure de traitement des messages d'une classe de fenêtres reçoit le message et l'identificateur de la fenêtre cible du message. Ces procédures sont en fait une immense construction « switch() ... case ». Chaque message est identifié par une constante qui apparaît au niveau d'un « case ». Il existe environ 250 messages actuellement définis dans Presentation Manager.
Si une procédure ne veut ou ne peut traiter un message particulier, elle peut laisser Presentation Manager s'en charger. Une des librairies (DLL) de Presentation Manager contient toutes les procédures de défaut de traitement des messages. En général, ces procédures se contentent de rendre la main, sans faire quoi que ce soit.
Presentation Manager traite deux types de queues de messages : celui relatif au système et celui relatif aux applications. Lorsqu'un événement se produit, Presentation Manager le transforme en message et l'inscrit dans la queue du système. Un module de Presentation Manager — le routeur — se charge de répartir ces messages dans les différentes queues des applications. C'est l'application qui était activée au moment où le message a été généré qui reçoit le message dans sa queue.
Chaque application est responsable de la création de la queue qui recevra les messages envoyés par Presentation Manager. Cette création se fait toujours au début du programme principal — la fonction main() — de l'application. C'est la fonction « WinCreateMsgQueue() » qui fait ce travail. Elle accepte deux arguments : le premier est l'identificateur de l'application et le second la taille de la queue.
Chaque application est identifiée par Presentation Manager grâce à un identificateur (étonnant, non !) obtenu avant toute autre chose par la fonction « WinInitialise() ». Celui-ci est employé par bon nombre de fonctions et permet à Presentation Manager de déterminer l'application qui fait appel à ses ressources.
En fin d'application, il faut détruire toutes les ressources utilisées. « WinDestroyWindow() », « WinDestroyMsgQueue() » et

les noms de fonctions de OS/2 et PM

Il y a une quantité impressionnante de fonctions dans OS/2 et Presentation Manager. Pour que les programmes restent lisibles, Microsoft a opté pour un système de noms de fonctions très significatifs : un préfixe de trois lettres et un suffixe dont la taille dépasse parfois 20 caractères.
Le préfixe désigne la catégorie de fonction. Le suffixe décrit l'objet de la fonction. Ce suffixe est d'un seul tenant, mais constitué de plusieurs mots. Pour rendre lisible ce nom qui constitue en fait une phrase, chaque mot commence par une lettre majuscule alors que le reste est en lettres minuscules. Par exemple, « VioGetDeviceCellSize » est une fonction qui fait partie du système de gestion d'écran (préfixe Vio) et qui interroge ce système pour obtenir (Get) la taille des cellules (CellSize) utilisées pour représenter un caractère pour un périphérique donné (Device).

Voici la liste des préfixes :
« Dos » désigne les fonctions de gestion des fichiers et périphériques au sens large,
« Gpi » est réservé aux primitives graphiques,
« Kbd » désigne les fonctions servant à la gestion du clavier,
« Mou » pour la gestion de la souris,
« Vio » sert à la gestion du système vidéo,
« Win » est utilisé pour les fonctions de gestion des fenêtres.

« WinTerminate() » se chargent respectivement de détruire les fenêtres, les queues de messages et l'application elle-même.
Nous avons maintenant toutes les cartes en main pour tracer la trame d'une application simple. Dans le programme qui nous sert d'exemple, il n'y a qu'une seule fenêtre et qu'une seule classe de fenêtre. La procédure qui traite les messages de ladite classe est « ClientWndProc() ».
Voici donc, en version simplifiée, le code de l'application :

```
main()
{
  WinInitialise();
  WinCreateMsgQueue();
  WinRegisterClas ( ClientWndProc );
  WincreateStdWindow();

  while (WingetMsg())
  WindispatchMsg();

  WindestroyWindow();
  WinDestroyMsgQueue();
   WinTerminate();
}

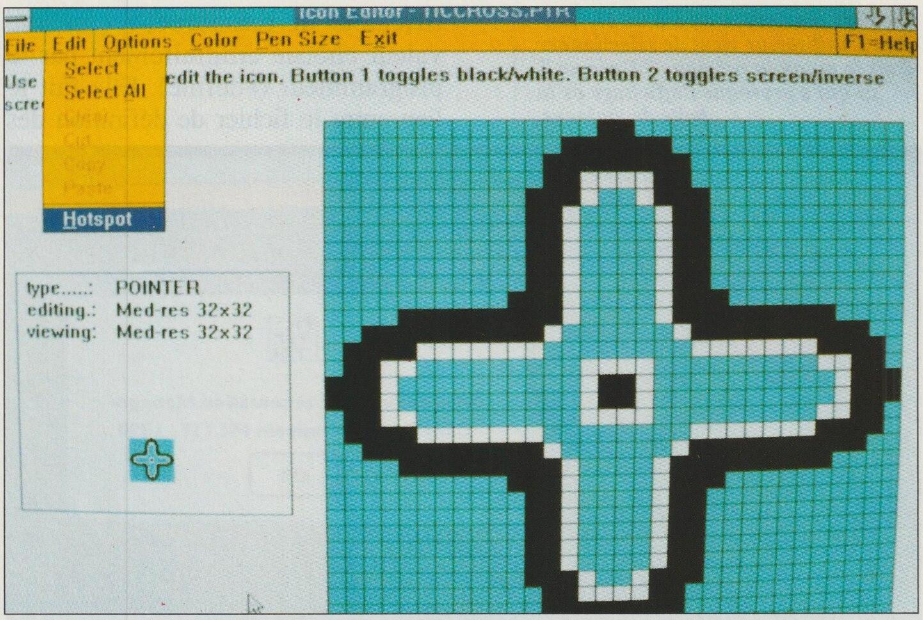
ClientWndProc( message )
{
  switch (message) {
  case WM_CREATE;
     .    .
  case WM_PAINT:

  }
  return(WinDefWindowProc(message));
}
```

Dans cette trame, aucune variable n'a été mentionnée. Toutes les fonctions prennent des arguments, seuls ceux significatifs pour mettre en évidence la trame ont été indiqués.

icones et pointeurs

Une application Presentation Manager se présente à l'écran sous forme d'au moins une fenêtre. Si l'utilisateur le désire, il peut la minimiser en la transformant en une



Fenêtre maximisée de l'éditeur d'icônes et bitmaps montrant un des pointeurs de la souris pour Tic-Tac-Toe. Ce qui apparaît en bleu sera transparent, ce qui est noir ou blanc le sera effectivement. Dans la petite fenêtre sur le côté, on voit le pointeur en vraie grandeur

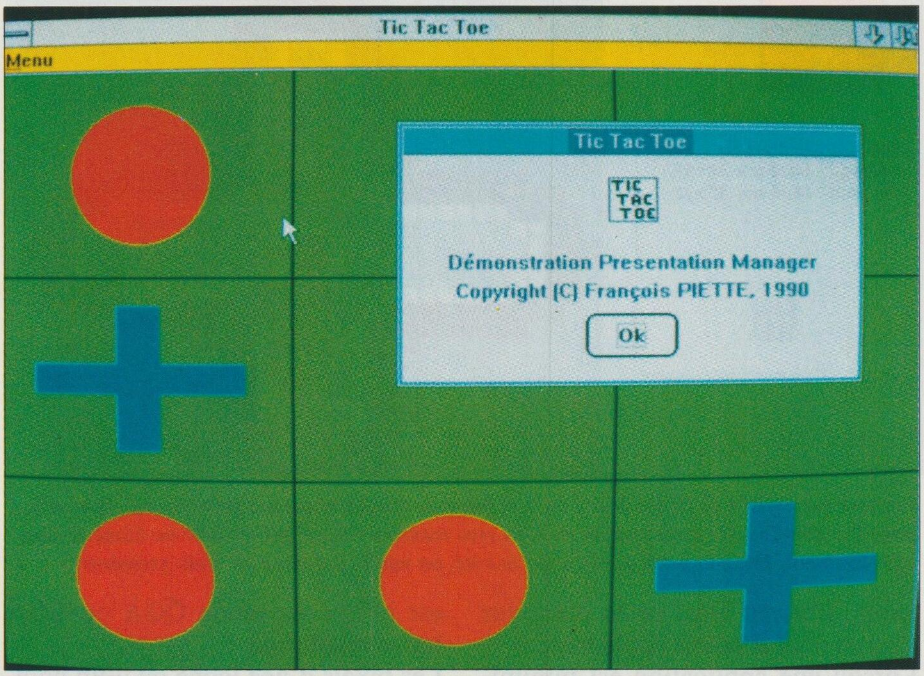
icone, libérant ainsi de la place sur l'écran pour d'autres applications.
Lorsqu'une application est minimisée sous forme d'icone, elle continue à fonctionner normalement, sauf qu'elle ne reçoit plus de message de l'utilisateur. Cet état dure jusqu'à ce que l'utilisateur demande à Presentation Manager de recréer la fenêtre sur l'écran.
Le pointeur de la souris se présente par défaut sous la forme d'une petite flèche, ce qui n'est pas forcément approprié à toutes les applications. Aussi le programmeur peut-il spécifier d'autres formes plus significatives et dépendantes du contexte. Dans notre exemple, il y a deux types de pointeurs selon la phase de jeu : une croix ou un rond.
Les icônes et les pointeurs sont de simples images bitmap (terme anglais désignant une matrice de points). Selon la résolution de l'écran, Presentation Manager permet trois résolutions pour les pointeurs : 64 x 64, 32 x 32 et 32x16 pixels. La dimension 32 x 32 est la plus courante, elle est utilisée avec les écrans EGA, VGA et Hercules. La plus haute résolution est employée avec d'autres types d'écran à haute résolution. Tandis que le format 32 x16 est utilisé en mode CGA ou EGA basse résolution.
Les pixels d'une icone ou d'un pointeur sont constitués de deux images bitmap : une servant de masque (ET, AND) et l'autre d'inverseur (OU EXCLUSIF, XOR). Avec cette construction, un point peut prendre quatre états différents :

AND	XOR	Effet
0	0	Point noir
0	1	Point blanc
1	0	Point transparent
1	1	Point inversé

La création des icônes et pointeurs est réalisée grâce à un programme interactif : ICONEDIT.EXE.

Pour utiliser les pointeurs et les icônes au sein du programme, il faut préalablement les charger. Cela est pris en charge par la procédure de traitement des messages (ClientWndProc) qui est associée avec les fenêtres concernées. C'est la fonction « WinLoadPointer() » qui réalise ce travail, lors du traitement du message obtenu lorsque Presentation Manager crée la fenêtre (WM_CREATE).
Chaque icône ou pointeur est identifié par deux nombres. Le premier, l'iden-

Fenêtre maximisée du jeu de Tic-Tac-Toe. L'utilisateur vient de sélectionner dans le munu la rubrique "A propos", ce qui a provoqué l'affichage de la boîte de dialogue.



tificateur de ressource, reçoit une valeur choisie arbitrairement par le programmeur (#define). Elle fait le lien entre le fichier de définition des ressources (fichier .RC) et le source (fichier .C). Le second est un identificateur renvoyé par la fonction « WinLoadPointer() ». Il est utilisé subséquemment chaque fois que l'on veut manipuler le pointeur ou l'icône. Presentation Manager sait quelle icône utiliser lorsque l'on minimise la fenêtre : c'est celle dont l'identificateur défini par le programmeur est identique à celui passé comme huitième argument de « WinCreateStdWindow() » lors de la création de la fenêtre.
Pour changer le pointeur de la souris, il suffit d'appeler la fonction « WinSetPointer() » avec l'identificateur du pointeur voulu. Cela se fait normalement lors du traitement du message indiquant un déplacement de la souris (WM_MOUSEMOVE).

les menus

Les menus sont des fenêtres spéciales, manipulées par Presentation Manager seul, qui contiennent des listes d'objets. Ces objets peuvent être de simples textes, des images bitmap ou des dessins tracés par l'application.
Les menus permettent à l'utilisateur d'effectuer un choix parmi les objets qui lui sont présentés. Ce choix, il peut l'effectuer au moyen du clavier ou de la souris.
Quand l'utilisateur fait un choix dans un menu, la procédure de traitement du menu — interne à Presentation Manager — envoie au « propriétaire » du menu un message qui l'identifie.
Les menus de Presentation Manager se présentent sous deux aspects : une barre de sélection et des sous-menus déroulants. Ce type d'interface est devenu très commun et probablement bien connu du lecteur. Nous ne la décrirons pas ici.
En règle générale, le programmeur définit un menu en passant par l'intermédiaire du compilateur de ressources (RC) qui reçoit un fichier texte décrivant le menu et qui génère un module objet qui aboutira dans le programme exécutable grâce à l'éditeur de liens (LINK).

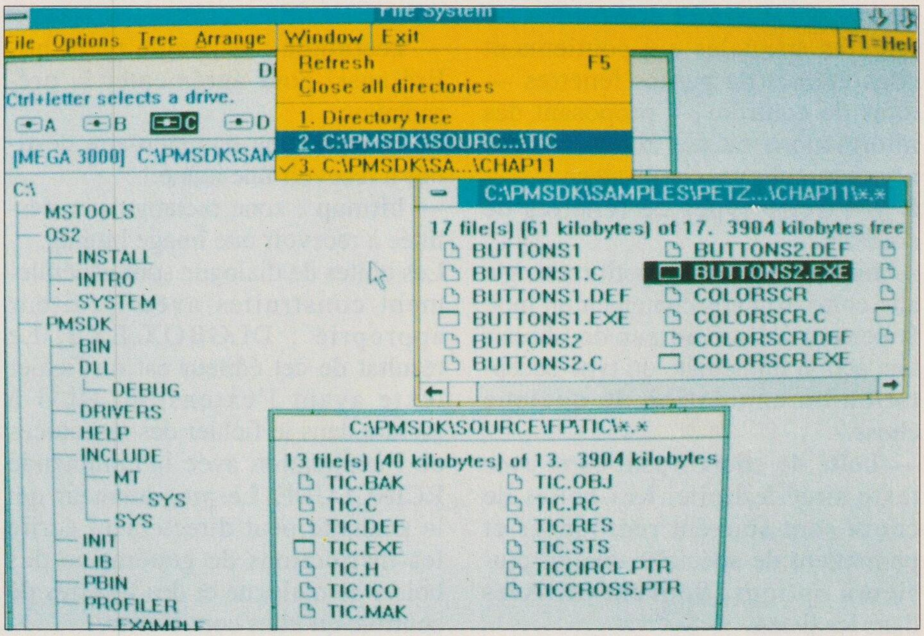
les noms des variables

Pour rendre le plus lisible et le plus parlant possible les programmes sources, Microsoft a décidé d'utiliser une convention simple dans les noms des variables. Cette convention prévoit qu'une variable possède un nom significatif de sa raison d'être et un préfixe indiquant son type de donnée. La première lettre du nom est écrite en majuscule, le préfixe et le nom sont accolés.
Les types de données utilisés sous Presentation Manager sont tous définis dans os2.h et s'écrivent en lettres majuscules.

Préfixe	Description	Exemple
f	indicateur booléen	BOOL fSucces;
ch	caractère sur 8 bits	CHAR chChar;
s	entier 16 bits, signé	SHORT sRate;
l	entier 32 bits, signé	LONG lDistance;
uch	caractère sur 8 bits, non signé	UCHAR uchScan;
us	entier 16 bits, non signé	USHORT usHeight;
ul	entier 32 bits, non signé	ULONG ulWidth;
b	entier 8 bits, non signé	BYTE bAttribute;
sz	chaîne de caractères	CHAR szName[];
fb	tableau de bits dans un BYTE	BYTE fbMask;
fs	tableau de bits dans un SHORT	SHORT fsMask;
fl	tableau de bits dans un ULONG	ULONG flMask;
sel	sélecteur de segment	SEL selSegment;
p	pointeur far (32 bits)	PCH pchBuffer;
np	pointeur near (16 bits)	NPCH npchBuffer;
a	tableau	CHAR achData[3];
i	index d'un tableau	USHORT ichIndex;
c	compteur	USHORT cb;
hf	identificateur	HFILE hf;
off	offset	USHORT offData;
id	identificateur	idSession

Le fichier texte qui décrit un menu est de structure assez simple. A la base, il y a l'instruction « MENUITEM » qui associe une chaîne de caractères — celle qui apparaît dans le menu — avec un identificateur — un nombre entier transmis dans le message généré par l'utilisateur lorsqu'il choisit cet élément.
Au-dessus de ces « MENUITEM » se trouvent les instructions « SUBMENU » et « MENU » qui définissent les niveaux supérieurs. Le tout étant assemblé en forme de bloc au moyen des instructions BEGIN et END ou les accolades ouvertes et fermées, au choix du programmeur.
Ci-dessous, le menu de notre application : une barre de sélection avec un seul élément (Menu) déroulant comportant lui-même trois éléments.
Le caractère tilde (~) sert à désigner les lettres qui activeront les options au moyen du clavier.
L'identificateur ID_TIC qui apparaît à droite de l'instruction MENU est l'identificateur général de la fenêtre principale — et unique — de notre application. C'est le fameux huitième argument de « WinCreateStdWindow() ». C'est cet identificateur commun qui permet à Presentation Manager d'associer la fenêtre, le menu et l'icône de l'application.
Les autres identificateurs (IDM_XXXX) sont purement arbitraires. Ce sont eux que Presentation Manager placera dans les messages transmis lorsque l'utilisateur choisira les éléments qui y correspondent, et ce sont eux que l'on retrouvera dans le listing au niveau de la construction « switch... case » qui traitera ces messages.
Précisément, lorsque l'utilisateur choisit un des éléments d'un menu, la procédure de traitement des messages de la fenêtre à qui appartient le menu recevra le message « WM_COMMAND ». Dans ce message, se trouve l'identificateur de l'élément de menu choisi. Voici un extrait simplifié et symbolique du code source C correspondant :

Vue d'un Gestionnaire de fichiers de OS/2-PM. La fenêtre principale montre l'arborescence du disque tandis que les deux petites fenêtres montrent deux sous-répertoires. La copie d'un fichier d'un répertoire dans l'autre se fait simplement en "traînant" le nom du fichier d'une fenêtre à l'autre.




```
MENU ID_TIC
BEGIN
    SUBMENU "~Menu", IDM_MENU
        BEGIN
            MENUITEM "~A propos...", IDM_ABOUT
            MENUITEM "~Efface",     IDM_CLEAR
            MENUITEM "~Quitte",     IDM_QUIT
        END
    END
```

```
ClientWndProc(msp, mp1, mp2)
{
        switch (msg) {
        case WM_CREATE:
                .
                .
        case WM_COMMAND:
                switch (COMMANDMSG(&msg)->cmd) {
                case IDM_ABOUT:
                        WinDlgBox();
                        return(0);
                case IDM_CLEAR:
                        .
                        .
                }
        }
}
```

fenêtres et boîtes de dialogue

Une boîte de dialogue et une fenêtre de dialogue sont deux choses identiques. Ce sont des formes de fenêtres spéciales qui contiennent elles-mêmes de petites fenêtres — dites de contrôle — proposant des informations ou des zones de données à remplir par l'utilisateur.

Il y a treize types de fenêtres de contrôle :

— **bouton poussoir** : petit rectangle aux coins arrondis contenant un mot. Il permet à l'utilisateur de choisir une action immédiate du type acceptation ou annulation de quelque chose.

— **boîte de choix** : petit carré avec texte situé à droite. Les boîtes de choix sont souvent regroupées et permettent de spécifier une ou plusieurs options. Souvent utilisées dans les listes.

— **bouton radio** : petit cercle avec texte informatif situé à droite. Ils sont généralement utilisés en groupe et permettent un choix exclusif d'une chose parmi l'ensemble, comme les présélections d'une radio.

— **barre de défilement horizontal** : rectangle fortement allongé avec un curseur que l'utilisateur peut faire glisser. Généralement associée à une autre fenêtre contenant des informations que le déplacement du curseur fait défiler.

— **barre de défilement vertical** : identique à la barre horizontale, mais placée dans l'autre sens évidemment.

— **boîte à liste** : rectangle avec une barre de défilement sur la droite. Une boîte à liste contient en général une liste d'éléments — par exemple une liste de fichiers — que l'utilisateur peut parcourir pour sélectionner un élément de la liste.

— **éditeur** : rectangle dans lequel l'utilisateur peut introduire un texte et l'éditer.

— **boîte de groupage** : rectangle avec un titre dans sa partie supérieure. Généralement utilisée pour grouper un certain nombre d'autres fenêtres de contrôle.

— **texte** : simple texte sans autre utilité que documentaire.

— **cadre** : simple rectangle utilisé pour encadrer d'autres fenêtres de contrôle et agrémenter la présentation.

— **rectangle** : rectangle plein utilisé aussi pour agrémenter la présentation.

— **icône** : zone rectangulaire destinée à recevoir une icône.

— **bitmap** : zone rectangulaire destinée à recevoir une image bitmap.

Les boîtes de dialogue sont généralement construites avec l'éditeur approprié : DLGBOX.EXE. Le résultat de cet éditeur est un fichier texte ayant l'extension .DLG à inclure dans le fichier des ressources de l'application avec la commande RCINCLUDE. Le programmeur qui le souhaite peut directement écrire les instructions de génération des boîtes de dialogue et des fenêtres de contrôle qu'elles contiennent.

Notre exemple de programme contient une boîte de dialogue appelée « A propos » parce qu'elle présente le programme. Voici les lignes qui la définissent dans le fichier de ressources.

```
DLGTEMPLATE ID_TIC
BEGIN

    DIALOG "Tic Tac Toe", ID_TIC, 80, 100, 200, 80,

        FS_NOBYTEALIGN ¦ FS_DLGBORDER ¦ WS_VISIBLE ¦
        WS_CLIPSIBLINGS ¦ WS_SAVEBITS,
        FCF_TITLEBAR

    BEGIN

        ICON ID_TIC
            -1, 90, 56,  0, 0
        CTEXT "Démonstration Presentation Manager"
            -1,  0, 38, 200, 8
        CTEXT "Copyright (C) François PIETTE, 1990"
            -1,  0, 28, 200, 8
        DEFPUSHBUTTON "Ok"
            DID_OK, 80,  8, 40,  16, WS_GROUP

    END
END
```

L'instruction « DLGTEMPLATE » crée une boîte de dialogue. L'instruction « DIALOG » définit la position, les dimensions, le titre et le style de la fenêtre de dialogue, elle-même composée d'une série de fenêtres de contrôle. Dans notre exemple, il y a une icône, deux textes centrés et un bouton poussoir. Pour chacun, la position, les dimensions, le titre éventuel et le style sont définis. Il sort du cadre limité de cet article de décrire en détails ces instructions. Les détails se trouvent dans le manuel « Ressource Compiler » intégré au kit « Microsoft OS/2 Presentation Manager Softset ».

Pour faire apparaître une fenêtre de dialogue, il suffit d'appeler la fonction « WinDlgBox() ». Un des arguments de cette fonction spécifie la procédure de traitement des messages reçus par la fenêtre. Dans notre exemple, il s'agit de « AboutDlgProc() ».

Le traitement des messages d'une boîte de dialogue est tout à fait similaire à celui d'une fenêtre ordinaire. Dans notre Tic-Tac-Toe, il est particulièrement simple puisqu'il

LOGICIELS DERNIERES VERSIONS POUR PC, XT, AT, COMPATIBLES, UNIX, XENIX ET MACINTOSH AUX

MEILLEURS PRIX

SOFTWORLD

17, Avenue Emile Zola, 75015 Paris. (Métro Javel)
Téléphone : (1) 40 59 02 99 Fax : (1) 45 79 95 55

		Prix Public TTC	Notre prix TTC
BASES DE DONNEES			
4ème Dimension	f	7353	5885
dBFast/Mac Compilateur dBase	a	3997	2950
FoxBase Mac Mono	æ	6820	5550
Omnis 5 *Blyth*	f	7110	6399
PAO / DESSIN			
Page Maker 4.0 *Aldus*	f	9607	7600
Canvas 2.0 *Deneba*	f	4140	3490

MATHEMATICA

Un outil intégral de calcul, de représentation graphique 3D et de programmation symbolique interactive de haut niveau pour Scientifiques, Mathématiciens. Version Mac ou autres systèmes
Version 1.2 Mac : Standard 7353,20 Frs TTC
Version 1.2 Mac : Avancée 11800,70 Frs TTC

		Prix Public TTC	Notre prix TTC
TABLEURS			
Excel 2.2 *Microsoft*	f	4732	3395
WINGZ Avec graphe 3D	f	5325	4650
EDITEURS DE TEXTE			
Word 4 *Microsoft*	f	3546	2650
Formulator *(Mathématiques)*	a	1950	1490
INTEGRES			
Works *Microsoft*	f	2953	2250
LANGAGES			
Think C 4.0	a	2953	**1995**
Think Pascal 3.0	a	1779	**1495**
Mac SchemeLisp+Tools.	a	nc	4250
Prolog 2.0 *Adv.A.I. Syst.*	a	3530	2950
Professional Programmer's Ext.	a	nc	2995
Prototyper (Générateur de Codes sources)	a	nc	1995
QuickBasic *Microsoft*	a	1175	895
Smalltalk V/Mac *Digitalk*	a	3084	2195
Turbo Pascal *Borland*	f	1180	850
Turbo Pascal DataBase	a	1180	850
Turbo Pascal Numeric	a	1180	850
Turbo Pascal Tutor *Borland*	a	1180	850
ZBasic 5.0 *Zedcor*	a	2022	1495
SCANNER ET O.C.R.			
Omnipage (O.C.R.)	a	11563	9750
Scanner couleur *Microtek*	a	29532	24900
UTILITAIRES			
PC Tools/Mac *Central Point*	a	995	695
SAM 2.0 (Antivirus)	a	1175	990
Virtual 030 pour Mac 68030	a	2360	1895

INGENIERIE

ACNAP Analyseur circuits AC	a	nc	4550
DCNAP Analyseur circuits DC	a	nc	2600
MICRO-CAP III (PC seulement)	a	nc	14950
SPP Traitement des signaux	a	nc	1950
Systat Statistiques-Graphiques	a	nc	7950

MUSIQUE			
MacRecorder *Farallon*	a	nc	2150
Professional Composer *Mark Uni.*	a	6227	4950

a : Version en américain.
f : Version en français.
æ : en américain avec possibilité d'échange en français.
nc : non communiqué
TOUS NOS PRIX SONT TTC
PRIX INDICATIF AU 22 JUIN 1990
Extrait de notre catalogue
Catalogue gratuit sur demande

		Public TTC	Notre TTC
ADA et Librairies			
Janus ADA Compiler DOS	a	nc	3500
Janus ADA Compiler 386	a	nc	5950
ASM et Librairies			
386 ASM/386 LINK*Pharlap*	a	nc	5595
Btrieve OS/2 5.00C*Novell*	a	nc	5950
EMS Toolkit *Realtime H.*	a	nc	1350
Intel ASM-386 Macro ASM	a	nc	6000
Intel ASM-86 Soft.Dev.Kit	a	nc	7500
MASM 5.1 *Microsoft*	a	1767	1350
PCX Effect + Source *Genus*	a	nc	1990
PCX Prog's Tlkit+Code Srce	a	nc	5990
PCX Text + Source *Genus*	a	nc	2990
BASIC et Librairies			
Basic 7.0 *Microsoft*	a	4732	3795
Basic Develop. Tools *Sterli.*	a	nc	1175
Btrieve *Novell*	a	2906	2550
FABS PLUS *Computer C*	a	nc	2350
GSS Graphics Dev.ToolDOS	a	nc	5950
Laser Pak Pro *Cresent*	a	nc	1495
Logic Gem *Sterling Castle*	a	nc	2350
MicroHelp Toolbox *MicroH.*	a	nc	1195
PDQ *Cresent*	a	nc	895
Power Screen *Blaise Comp.*	a	nc	1490
MicroHelp Utility *MicroHelp*	a	nc	850
PCX Effects + Source *Genus*	a	nc	1990
PCX Prog's Tlkit+Code Srce	a	nc	5990
PCX Text + Source *Genus*	a	nc	2990
ProMath *Hammerly*	a	nc	1395
Quick Basic 4.5 *Microsoft*	f	1175	895
QBase *Crescent*	a	nc	1495
QuickPak Pro *Crescent*	a	nc	1495
QuickPak Scientific *Crescent*	a	nc	950
QuickScreen *Crescent*	a	nc	895
Screen Machine *MicroHelp*	a	nc	850
Scientific SubroutineLibrary	a	nc	2250
Stay-res + *MicroHelp*	a	nc	1295
Turbo Basic *Borland*	æ	1185	820
Turbo Basic Toolbox chaque	æ	1180	820
True Basic *True basic*	a	nc	995
ZBasic *Zedcor*	a	nc	895
C, C++ et Librairies			
C Asynch Manager *Blaise*	a	nc	1890
C Compiler 6.0 *Microsoft*	a	5325	3895
C Tools Plus *Blaise Comput*	a	nc	1495
GFX Graphics Library	a	nc	2135
GFX Fonts & Menus Lib.	a	nc	2135
GFX Font Editor	a	nc	949
GFX Screen Dump	a	nc	949
GreenLeaf Data Windows	a	nc	4250
Halo *Media Cybernetics*	a	nc	3930
High C 386 *MetaWare*	a	nc	10500
Lattice C *Lattice*	a	5812	3350
Quick C 2.0 *Microsoft*	a	1767	1395
Super Functions *Greenleaf*	a	nc	2950
Turbo C 2.0 *Borland*	f	1773	1250
Turbo C Pro 2.0 *Borland*	f	3552	2350
Turbo C Tools *Baise Comp*	a	2123	1495
Turbo C++ *Borland*	f	2366	1650
Zortech C++ 2.0 *Zortech*	a	2366	1895
Zortech C++ 2.0 Developper	a	4738	3895

		Public TTC	Notre TTC
COBOL			
Cobol 3.0 *Microsoft*	a	8290	6195
Cobol / 2 *Microfocus*	a	nc	18000
Cobol spII *Flexus*	a	nc	4795
DEBUGGEURS			
Intel ASM-386 Soft.Dev.	a	nc	7500
Periscope IV (25 MHz)	a	nc	25950
Pfix86+ *Sage*	a	nc	2950
View232 *Blaise Computing*	æ	nc	1890
EDITEURS			
Brief/dBrief *Solution Systems*	a	4459	3630
EC Editor *C Source*	a	nc	1126
Epsilon *Lugaru Software*	a	nc	1950
Norton Editor *Peter Norton*	æ	nc	750
VEdit Plus *CompuView*	a	nc	1850
GENERATEURS			
Automated Programmer*KGK*	a	nc	4950
Matrix Layout pr Basic, C, Pascal	a	nc	1950
PCYACC Corporate*Abraxas*	a	nc	5500
INTELLIGENCE ART.			
PC SCHEME *Texas Instru.*	a	nc	1995
Smalltalk V *Digitalk*	a	nc	895
Smalltalk V/286 *Digitalk*	a	nc	1450
Communication Pak *Digitalk*	a	nc	425
EGA/VGA Pak *Digitalk*	a	nc	425
Goodies 1 ou 2 ou 3 *Digitalk*	a	nc	425
µuLisp *Soft Warehouse*	a	nc	3558
µuLisp+Compiler *Soft Ware.*	æ	nc	4744
PASCAL/LIBRAIRIES			
MTASK pro *Ramsi*	f	2313	**1950**
Pascal 4.0 *Microsoft*	a	4732	3695
Pascal Asynch Man.*Blaise C.*	a	nc	1750
Pascal Tool 1 *Blaise C.*	a	nc	1250
Pascal Tools 2 *Blaise C.*	a	nc	995
Pascal View 232 *Blaise C.*	a	nc	1890
Quick Pascal *Microsoft*	f	1720	**1395**
Turbo Pascal 5.5 *Borland*	f	1773	**1250**
Turbo Pascal Pro 5.5	f	3552	**2350**
AUTRES LANGAGES			
Actor *White Water Group*	a	nc	6950
Fortran 5.0 *Microsoft*	a	5990	4450
Guideline C++ *Guidelines*	a	nc	3950
DIVERS			
Btrieve *Novell* (Gestion Fichiers)	a	nc	2550
Merge 386 2 Utilisateurs *Locus*	a	4151	4000
Norton Guides *Norton* Chque	a	1186	995
PC MOS/386 Mono *Soft.Link*	a	2953	2495
Hoops/386-32 *Ithaca*	a	nc	7950
Window 3.0 *Microsoft*	f	2360	1750
Windows 286 + Souris	f	3665	2595
Windows 386 + Souris	f	4851	3495
Windows Devel. Toolkit 3	a	5918	4450

UNIX
Systèmes / Langages / Fichiers / SGBD
Editeurs de Texte / Utilitaires
Nous contacter : (1) 40 59 02 99

		Public TTC	Notre TTC
BASES DE DONNEES			
dBFast PLus Compiler dBaseIII+	a	3997	3250
dBFast/Window	a	3997	3250
dB XL *Wordtech systems*	f	4715	3850
Fox Base+ 2.10	f	9429	7500
Genifer+Template CompilerFoxBase	a	nc	4195
Paradox 3.0 *Borland*	f	9963	6950
Super DB *Computer Asso.*	f	7021	5200
CAO / DAO / PAO			
Design CAD 2D (3D +1000 Frs)	f	4685	3950
Page Maker 3.0 PC *Aldus*	f	9610	6750
Ventura 2.0 *Rank Xerox*	f	9192	8550
COMPTA/GESTION			
Ciel! Compta-Gestion *Ciel!*	f	1156	925
Compta Saari Major *Saari*	f	16905	13950
INTEGRES			
Ability Plus Dual *Migent*	f	1927	1695
Works PC 2.0 *Microsoft*	f	2950	2050
TABLEURS			
Compile 12C Compiler WK.	a	nc	8950
Excel PC (286 et 386)	f	5918	3895
Lotus 123 Version 3.0	a	6750	4950
Multiplan 4 *Microsoft*	f	3309	2395
Quattro Pro *Borland*	f	5924	4150
VP Planer + 3D *Paperback*	f	3440	2595
EDITEURS DE TEXTE			
Chiwriter (Scientifiques)	f	1235	1195
MathEdit (Mathématiques)	a	2366	**2366**
Sprint 1.5 *Borland*	f	2959	2050
Word pour Window	f	5918	4150
Wordperfect 5.1 *Wordperf.*	f	5693	4595
UTILITAIRES / SYST.			
386/DOS-Extender *PharLa*	a	nc	14950
386 MAX Pro *Qualitas*	a	1885	1495
Above Disc EMS 4.0	æ	1510	1195
CopyWrite *Quaid*	a	1175	695
Disk Explorer *Quaid*	a	1175	695
Norton Adv. Utilities	f	1779	1550
PC Tools Deluxe 6.0	f	2005	**1395**
Virusafe Plus	a	1175	**995**
MATHEMATIQUES			
Derive *Soft Warehouse*	a	nc	**2372**
Greenleaf Business MathLib	a	nc	3950
Grapher *(Graphiques scientif.)*	a	2953	2366
HiWire Plus Wintek	a	nc	9500
MathCAD 2.5 *MathSoft*	a	6274	4950
Mathematica DOS-386	a	nc	10318
Mathematica DOS-386/387	a	nc	14766
Mathematica DOS-386/Weitek	a	nc	19213
QuickPak Scientific *Crescent*	a	nc	895
Sciences & Engineer. Tools	a	nc	1195
Surfer *(Surface, carte géo. 3D)*	a	6111	4950
SYSTAT 5.0 + SYGRAPH	a	nc	10615
TKSolver	a	nc	4685

MATERIEL
Nous fournissons toute la gamme de produits de
HEWLET PACKARD
-20% sur le prix public

BON DE COMMANDE
SOFTWORLD
17, Av. Emile Zola, 75015 Paris. (Métro Javel)
Téléphone : (1) 40 59 02 99 / Fax : (1) 45 79 95 55

Mon ordinateur est du type : (Macintosh) (PC Compatible)
Format de disquette : 3 1/2 ○ ou 5 1/4 ○
Je veux commander les produits notés ci-contre.
Mode de paiement :
○ Par Chèque ci-joint.
○ Par Contre remboursement +40 Frs (montant d'achat <2000 Frs)
○ Par Carte Bleue N° ____
Date d'expiration : ____
Date : ____
Signature : ____ 1990

Nom : ____ Prénom : ____
Adresse : ____
Code postal : ____ Localité : ____
Téléphone : ____

Désignation	Quantité	Prix TTC

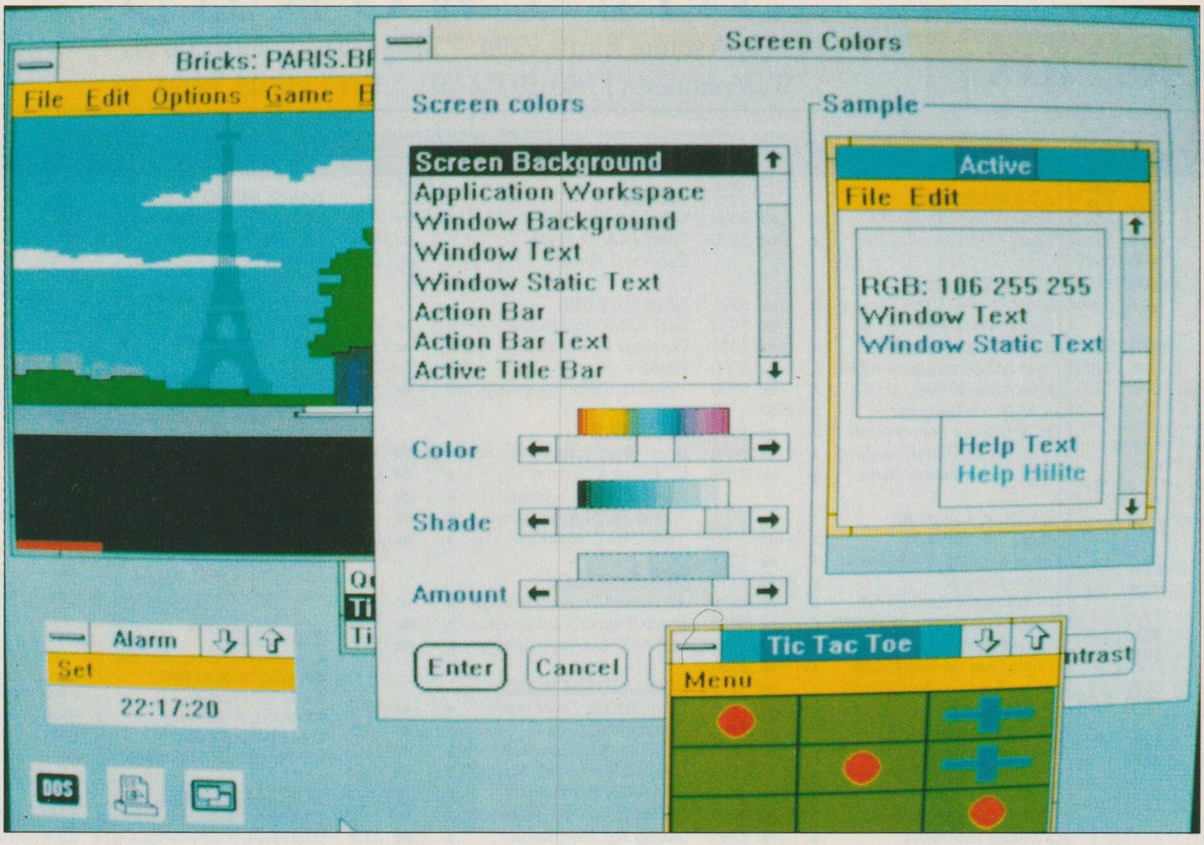
Frais d'envoi : ____
Total TTC : ____
Frais d'envoi recommandé obligatoire 40 Frs par 1000 Frs

Aucun règlement ne sera encaissé avant l'expédition des marchandises

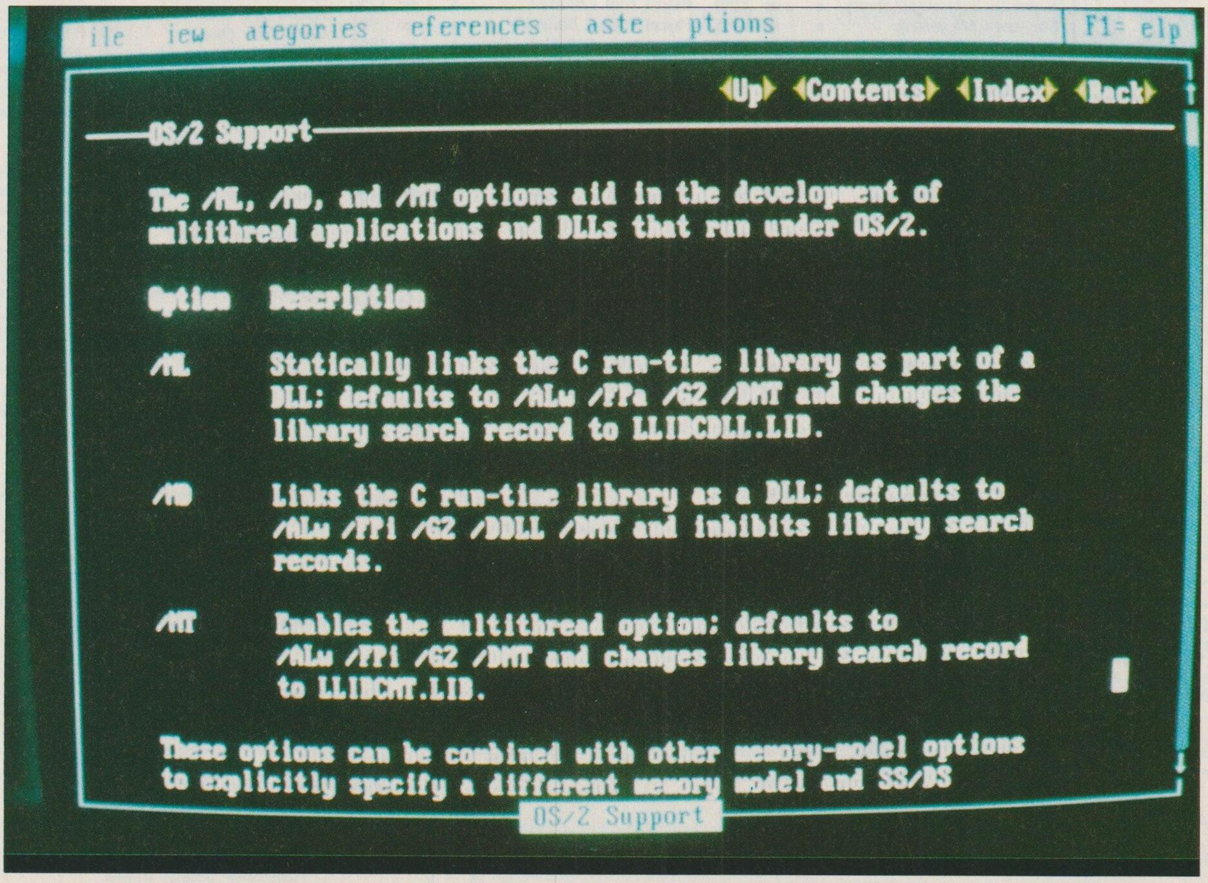
Nous fournissons la plupart des produits français et étrangers à des prix défiant toute concurrence et dans un délai record.
Plus de 2000 produits pour PC et MAC.
Applications verticales, Bases de données, D.A.O., Editeurs de Textes, Graphiques, Ingénierie, Intégrés, Langages, Mathématiques, Librairies, P.A.O, Statistiques, Tableurs, Utilitaires...

Heures d'ouverture
10h à 12h et de 14h à 19h
Fermé Samedi , Dimanche et Fériés

Cerclez info pub 21



Ecran de OS/2-PM montrant principalement une des fenêtres du panneau de contrôle, celle qui permet de choisir les couleurs qui apparaissent dans les fenêtres standard. Les autres fenêtres montrent un jeu de "casse-brique" avec la Tour Eiffel, un jeu de Tic-Tac-Toe, une horloge alarme et quelques icônes d'application tournant en arrière - plan.



Microsoft Quickhelp : écran obtenu en cliquant sur "OS/2 support" dans les options du compilateur.

consiste à supprimer la boîte dès réception du message qui correspond à l'appui sur le bouton « OK » que contient cette boîte. Cette suppression s'opère par l'intermédiaire de la fonction « WinDismissDlg() ».

le compilateur de ressources

Nous venons de le voir dans tout ce qui précède : une application Presentation Manager contient non seulement du code en langage C, mais aussi ce que l'on appelle des ressources : icônes, pointeurs, menus, boîtes de dialogue, images bitmap et polices de caractères.
Chacune de ces ressources est fondamentalement une structure de données que le programmeur pourrait initialiser dans son programme. Toutes les structures sont définies dans la documentation de Microsoft et rien, si ce n'est l'ampleur du travail n'empêche de réaliser tout cela « à la main ».
Heureusement, Microsoft fournit un outil très précieux : le compilateur de ressources (RC). Ce programme lit un fichier texte contenant sous forme symbolique une description des ressources. Nous en avons vu des exemples ci-dessus avec le menu et la boîte de dialogue. Le compilateur de ressources transforme le fichier texte en module objet que l'éditeur de liens combinera avec les modules objets produits par le compilateur C, pour créer le programme exécutable.
Le compilateur de ressources est secondé par trois autres programmes : l'éditeur de boîtes de dialogue, l'éditeur de polices de caractères et l'éditeur d'icônes. Ce dernier est utilisé tant pour les icônes que pour les pointeurs de la souris et les images bitmap en général.

les fonctions graphiques

Presentation Manager est très riche en fonctions graphiques : plus de 200 ! Le principe général qui a prévalu à la réalisation de ces fonctions est l'indépendance vis-à-vis du matériel. Tout est prévu pour que le programmeur puisse écrire des applications complètement indépendantes du matériel sur lequel elles fonctionnent, tant en matière d'écran que de table traçante ou d'imprimante.
Pour faciliter la lecture du texte qui va suivre, nous parlerons du système graphique avec l'appellation GPI qui est l'abréviation de « Graphic Programming Interface » et qui se trouve aussi être les trois lettres employées comme préfixe de toutes les fonctions graphiques.
On peut diviser le monde des graphiques en deux groupes distincts : les graphiques vectoriels et les graphiques point par point ou bitmap. GPI offre des fonctions pour exploiter les deux types de graphiques.
Le graphique vectoriel peut être représenté comme une table traçante à laquelle on transmet, par exemple, les coordonnées de deux points et qui trace seule la ligne qui les joint. Dans ce système, le matériel trace des figures géométriques sur la base des coordonnées de points caractéristiques, tels les extrémités d'une ligne, ou le centre et le diamètre d'un cercle.
Le graphisme point par point (ou bitmap) est celui des écrans graphiques des PC : il est constitué d'une matrice rectangulaire de points. Chaque point pouvant prendre une couleur choisie dans un ensemble fini. Par exemple, un écran VGA standard comporte 480 lignes de 640 points chacune. Chaque point pouvant prendre une couleur parmi 16. Un tel écran est bien adapté pour reproduire des images provenant, par exemple, d'un scanner. L'unité de travail étant le point dont on donne les coordonnées et la couleur.
Bien évidemment, il est possible d'écrire des programmes qui transforment une commande de tracé d'une ligne au départ des coordonnées de ces extrémités, en une série de commandes qui active les points qui justement font apparaître avec plus ou moins de précision la ligne en question. Une ligne oblique tracée de la sorte ne sera jamais qu'un escalier aux marches plus ou moins fines selon la résolution de l'écran employé.
Une caractéristique généralement commune aux systèmes bitmap est la possibilité de lire l'image présentée. Donc non seulement on peut allumer ou éteindre des points, mais on peut aussi interroger le système et connaître la couleur ou l'état de tel ou tel point. Il devient ainsi possible de déplacer ou de recopier des portions d'image.
Lorsque l'on parle de graphismes, il faut aussi évoquer le système de coordonnées, de résolution, de transformations d'image et de couleur.
GPI dispose de l'ensemble des possibilités mentionnées ci-dessus. Cependant, le programmeur devra quand même réfléchir un peu. Il n'existe pas de miracle : s'il est facile de transformer une commande de tracé de vecteur en une série de commandes de tracé de points, il est beaucoup plus difficile sinon impossible d'effectuer l'opération inverse. Cela signifie, par exemple, que le programmeur qui voudra offrir l'option d'imprimer sur table traçante devra éviter certaines fonctions impossibles à reproduire avec ce matériel à cause de sa structure.
Notre exemple de programme utilise seulement quelques-unes des fonctions de GPI, toutes relatives au système de tracé vectoriel. Voyons en détail la fonction qui dessine la croix.
L'algorithme du tracé de la croix est simple : choisir la couleur du tracé, tracer le polygone qui définit le contour de la croix, colorier l'aire définie par le polygone.
Pour bien comprendre le mécanisme du tracé, il faut voir la fenêtre de l'application comme une table traçante à laquelle on transmet des commandes qui déplacent sa plume. Ainsi, pour réaliser le contour, on appelle la fonction « GpiMove() » qui fixe les coordonnées du point où doit se poser la plume, puis on appelle « GpiPolyLine() » qui prend une liste de coordonnées de points et déplace la plume de la position cou-

rante (celle donnée par GpiMove) successivement vers les coordonnées indiquées dans la liste.
Pour effectuer le coloriage, il suffit d'indiquer à GPI le début et la fin d'une aire avec les fonctions « GpiBeginArea() » et « GpiEndArea() ». GPI colorie l'aire en question dans la couleur active à ce moment. « GpiSetColor() » se charge d'indiquer la couleur désirée.
Les fonctions « GpiMove() » et « GpiPolyline() » prennent en argument un point ou un tableau de points. Chaque point est représenté par une abscisse (X) et une ordonnée (Y). Le type de donnée POINTL est une structure à deux entrées : l'une pour la coordonnée X et l'autre pour Y.
La fonction essentielle de « DrawCross() » est de calculer les coordonnées des sommets du polygone qui représentent la croix. Celles-ci sont calculées proportionnellement au rectangle dont DrawCross reçoit les coordonnées (les quatre coins), de telle manière que la croix soit centrée et bien proportionnée. En agissant de la sorte, quelles que soient les dimensions de la fenêtre et la résolution de l'écran utilisé, les croix sont toujours proportionnellement identiques à elles-mêmes.
Toutes les fonctions graphiques utilisées ont comme premier argument un identificateur « d'espace de présentation ». C'est un nombre rendu par Presentation Manager, par exemple en appelant la fonction « WinBeginPaint() », qui identifie une zone de dessin. Grâce à cet outil, Presentation Manager peut s'occuper de ne dessiner que la partie visible des entités géométriques tracées par le programme (c'est le « clipping »).

l'interface avec le clavier

Dans notre jeu de Tic-Tac-Toe, l'utilisateur peut employer les flèches du clavier pour déplacer le pointeur vers la case où il désire placer sa croix ou son cercle. En appuyant sur la touche ENTER, il dépose le signe dans la case au-dessus de laquelle se trouve le pointeur.
Lorsque l'utilisateur appuie sur une touche du clavier, la fenêtre qui est sélectionnée à ce moment reçoit un message (WM_CHAR) indiquant qu'une touche a été pressée, et quelle est la touche en question.
Pour traiter cet événement, il suffit d'une part de demander à Presentation Manager la position actuelle du poin-teur (WinQueryPointerPos), d'au-tre part de convertir les coor-données obtenues relativement à l'écran (le « desktop ») en coordonnées relatives à la fenêtre du jeu (WinMapWindowPoints). Puis il faut calculer la nouvelle position en fonction de la tou-che enfoncée, convertir en sens inverse les nouvelles coordon-nées, puis déplacer le poin-teur au nouvel endroit (WinSetPointerPos).
S'il s'agit de la touche ENTER, le traitement est le même que si l'utilisateur avait cliqué sur le bouton gauche de la souris. Le moyen le plus simple de réaliser l'opération est d'envoyer à la fenêtre du jeu le même message que celui que Presentation Manager aurait envoyé si effectivement on avait appuyé sur le bouton. C'est la fonction « WinSendMsg() » qui se charge de ce travail.
Presentation Manager envoie un message WM_CHAR chaque fois que l'on appuie ou que l'on relâche une touche au clavier. Un certain nombre de touches sont « virtualisées », c'est-à-dire qu'elles ont un code qui est indépendant du type de clavier utilisé. Toutes les touches de fonction et le pavé de traitement de texte (flèches, etc.) sont « virtualisées ». Certaines touches sont qualifiées de « mortes », par exemple l'accent circonflexe qui ne génère pas de code en lui-même, mais qui modifie le code d'une touche s'il la précède : le code de la lettre E est différent du code de la lettre E plus accent circonflexe. Pour les touches ordinaires, le message contient son code ASCII ainsi que son SCAN-CODE. Un indicateur faisant partie du message précise la nature de l'événement ou de la touche.

outils de développement

Pour mener à bien la réalisation d'un programme comme le Tic-Tac-Toe, il est nécessaire de posséder d'une part le compilateur Microsoft C (au moins la version 5.1, la 6.0 si possible) et d'autre part, le « Presentation Manager Softset » qui se compose de manuels de référence et d'une série de programmes comme l'éditeur de liens (LINK), l'éditeur de boîtes de dialogue (DLGBOX), l'éditeur d'icônes (ICONEDIT) et le compilateur de ressources (RC). Dans notre exemple, ces programmes sont les seuls nécessaires, mais le kit en contient d'autres, utiles avec des programmes plus complexes ou dans des contextes différents.
L'application Tic-Tac-Toe se compose des fichiers suivants :

TIC.C : source du programme en langage C, créé avec un éditeur de textes,
TIC.H : fichier à inclure contenant des définitions, créé avec un éditeur de textes,
TIC.DEF : fichier de définitions pour l'éditeur de liens, créé avec un éditeur de textes,
TIC.ICO : fichier contenant la définition de l'icône du programme, créé avec ICONEDIT,
TICCIRCL.PTR : fichier contenant la définition du pointeur de la souris en forme de cercle, créé par ICONEDIT,
TICCROSS.PTR : fichier contenant la définition du pointeur de la souris en forme de croix, créé par ICONEDIT,
TIC.RC : fichier de définitions des ressources, créé avec un éditeur de textes,
TIC.MAK : fichier de commandes pour NMAKE, créé par le PWB (Programmer's Work Bench) qui accompagne le compilateur Microsoft C6.0, ou créé avec un éditeur de textes.
Le schéma ci-contre décrit le flux des fichiers et des programmes utilisés pour bâtir le fichier exécutable.

ADA LOG

Et pavé numérique

DACTYL'AID

Ne regardez plus votre clavier, utilisez vos dix doigts! DACTYL'AID permet de vous entraîner à votre rythme, même si vous tapez (mal) depuis des années.
Un chronomètre et un compteur de fautes permettent de mesurer vos progrès. Divers exercices sont proposés, vous pouvez utiliser vos propres fichiers comme modèles d'entraînement ou entrer les exercices d'une méthode.
Fonctionne avec tous les plans de clavier (éventuellement redéfinissables par l'utilisateur). Entraînement au pavé numérique.
DACTYL'AID est conçu pour les utilisateurs de micros qui veulent augmenter rapidement leur rendement sans s'astreindre à un entraînement de dactylo professionnelle.

Nouveau!

DISQUE-PILOTE

Un ensemble d'utilitaires (copie, répertoire, effacement etc.) utilisant des "méta-jokers" qui multiplient à l'infini les possibilités des commandes standard.
Un système de sauvegarde intelligente, qui permet de ne sauver que ce qui est nécessaire, juste quand c'est nécessaire, même sur disquette 360 Ko. Plus de sauvegardes interminables et que l'on hésite à effecturer!
Des programmes qui augmentent les possibilités des fichiers "BAT" (menus, questions) et un système de protection permettant de limiter l'accès à toute la machine, ou juste à quelques répertoires, par mot de passe.
Plus un mode d'emploi très complet qui vous explique comment organiser un disque! Le complément indispensable de votre disque dur.

Machine/PC

Transforme votre PC + imprimante en machine à écrire électronique perfectionnée.
Fonctions de justification, centrage, frappe au kilomètre, caractères gras et soulignés.
Beaucoup plus simple et rapide à utiliser qu'un traitement de textes.
Indispensable pour remplir des formulaires, ou taper du courrier rapidement. Ne gardez plus une machine à écrire à côté de votre micro.!

Offre spéciale

"Nouvel utilisateur"

Un élégant porte-documents toile
+ DACTYL'AID
+ MACHINE/PC
+ CLASSIFICHE
+ FACICOMPTES

~~1190 F.~~ TTC * 990 F.

* en 5,25". En 3,5": ~~1690F.~~ 1190 F. TTC

Non disponible dans les FNAC

MODIFICH

Un "patcheur" d'une grande simplicité d'emploi pour inspecter et modifier en mode plein écran n'importe quel fichier ou la structure des écran.
Indispensable pour voir la structure des fichiers générés par les applications ou récupérer des données endommagées.
Fonctions de recherche en hexadécimal ou ASCII.

VISITRAP

Vous voulez comprendre le mécanisme des interruptions MS/DOS? VISTRAP vous montre l'état de la machine et des registres en plein écran, avant et après l'interruption. Vous la voyez fonctionner directement au lieu d'explorer à tâton sous DEBUG.
Une fenêtre ouverte sur le système d'exploitation qui vous en apprendra plus que bien des livres.

Logiciels pour IBM/PC et compatibles

FACICOMPTES

Votre cahier automatique, pour suivre vos comptes sans vous embarasser d'une comptabilité d'entreprise!
Aide à la vérification des relevés bancaires avec recherche automatique d'erreurs. Répartition des dépenses en rubriques, gestion de chèques et cartes de crédit avec calcul automatique des dates de valeur. Totaux par rubrique et par opérateur. Entrée automatique d'écritures à date fixe.
Copie de sauvegarde " BAK", archivage des écritures vérifiées. Gère un nombre illimité de comptes, avec possibilité de transfert d'écritures entre comptes. Calculette disponible à tous moments.
Prêt à l'emploi, avec possibilité de tout redéfinir selon vos besoins. Parfait pour votre comptabilité personnelle, mais aussi pour gérer tout ce qui peut se compter.

Turbo Kit 4.0

Des composants logiciels pour Turbo Pascal®. Un ensemble de procédures prêtes à l'emploi, pour faire des masques de saisie et des menus, dessiner des cadres, gérer la date et l'heure, manipuler les noms de fichiers et les répertoires, faire de l'édition ligne et pleine page, accéder aux variables d'environnement, analyser les chaînes d'options et même lire et écrire en hexadécimal. Et tous les modules sont en français!
Avec Turbo-Kit, vos programmes auront le "look" professionnel.
Fourni avec en exemple un programme de gestion d'étiquettes et des utilitaires qui a eux seuls valent le prix de la disquette.
Une documentation papier complète explique les modules et donne une foule de conseils de programmation. Un véritable cours de belle programmation par l'exemple.
Turbo Kit 4.0 est fourni sous forme de sources d'"unit" pour Turbo Pascal 4.0 et plus (compatible 5.0 et 5.5)

CLASSIFICHE

Une gestion de fiches ultra-simple à utiliser, car imitant exactement ce que vous avez l'habitude de faire avec des fiches en carton. Aucune connaissance informatique n'est nécessaire. Les fiches sont toujours triées, et vous les retrouvez instantanément. Aucune limitation sur le contenu des fiches.
Il est possible de définir des fiches "pré-imprimées" pour fournir un modèle avec des rubriques prédéfinies. Sorties sur imprimante, avec sélection selon un modèle pour l'index. Possibilité d'échanger des données avec les autres logiciels.
Fourni avec un exemple d'utilisation en répertoire téléphonique.

COMBIEN ?

	5,25"	3,5"
DISQUE-PILOTE	390.00	440.00
DACTYL'AID	290.00	340.00
FACICOMPTES	290.00	340.00
MACHINE/PC..................	290.00	340.00
CLASSIFICHE	320.00	370.00
TURBO KIT 4.0	390.00	440.00
VISITRAP	190.00	240.00
MODIFICH	190.00	240.00
Kit "nouvel utilisateur" ...	990.00	1190.00

Prix TTC

En vente: dans les FNAC chez General Video

Cerclez info pub 22

Commande: Offre spéciale DISQUE-PILOTE DACTYL'AID FACICOMPTES TURBO KIT 4.0
...... CLASSIFICHE MACHINE/PC VISITRAP MODIFICH
...... Disquettes 5,25" Disquettes 3,5"
Nom: ..
Adresse: ..
..

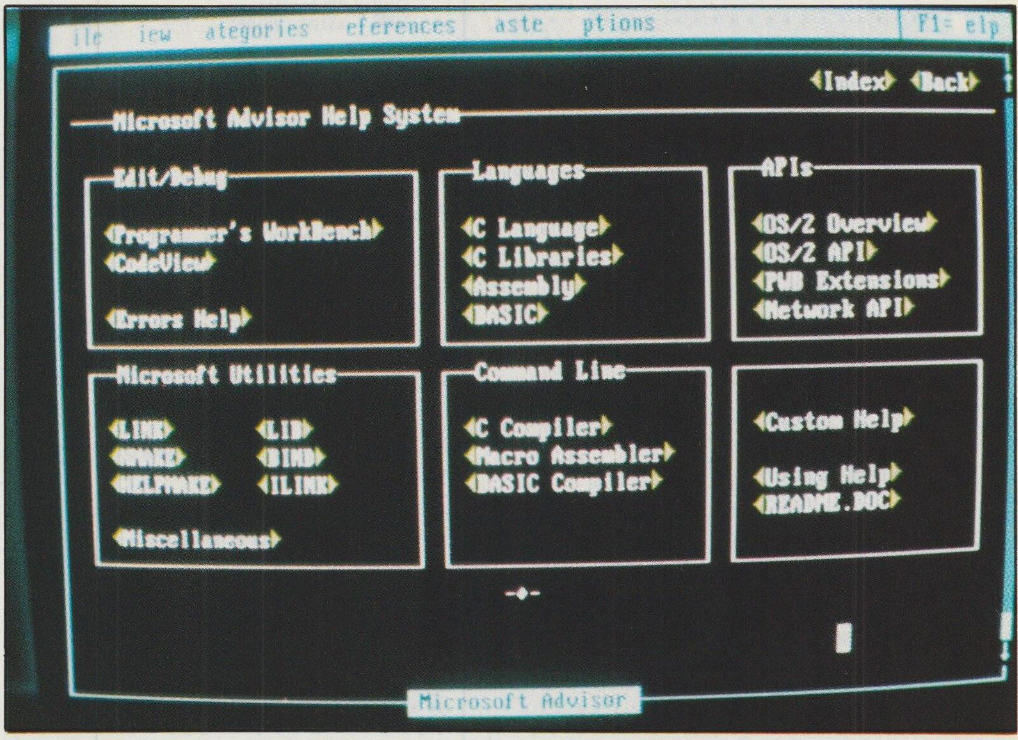
Bon à renvoyer avec votre réglement à: AdaLog, 115 av. du Maine 75014 Paris

```
|-===========|  ___________________________
|        | |        |
| EDITEUR |------>|   TIC.C   |-----------,
|        | |        |  |   |-===============|
|_===========|  |-------------------------  |  |        |
                 |------------>| COMPILATEUR |-------------,
|-===========|  _____________________________,  ----------->|    C   |  |
|        | |        |   |   |          |  |
| EDITEUR  |------>|   TIC.H  |-----------   |_===============|  |
|        | |        |
|_===========|  |-----------------------------                    |
                           |
|-===========|  ____________________________,                    |
|        | |        |            |
| ICONEDIT |------>|   TIC.ICO  |----------,                |
|        | |                    |
|_===========|  |-----------------------------  |                  |
                 |          |
|-===========|  ____________________________,  |  |-===============|  |
|        | |        |  |---------->|    |  |
| ICONEDIT |------>|TICCROSS.PTR|-----------------------------> | COMPILATEUR |-------,  |
|        | |        |  ---------->|   RC    | | |
|_===========|  |-----------------------------   |------->|    | |
                 | |  |_===============| | |
|-===========|  ____________________________,  | |             | |
|        | |        |  | |            | |
| ICONEDIT |------>|TICCIRCL.PTR|------------- |           | |
|        | |                     | |
|_===========|  |-----------------------------    |           | |
                 |          | |
|-===========|  ____________________________,  |           | |
|        | |        |            | |
| EDITEUR  |------>|   TIC.RC   |------------          | |
|        | |        |            | |
|_===========|  |---------------------------------------------------------------------  |
                 |     ____________
                 V        V
|-===========|   ___________________________,   __________,   -
______________________________________________,
|        | |        | |     | |        |
| EDITEUR  |------>|   TIC.DEF | |   TIC.RES  | |   TIC.OBJ   |
|        | |        | |     | |        |
|_===========|  |---------|-------------- |-----------|------------------ |-------|--------
              |-----------------,  |  __________________________
                 | | |
                 V V V
               |-===============|
               |  EDITEUR   |
               |  DE LIENS  |
               |   LINK    |
               |_===============|
                  |
                  V
               _____________________________,
               | EXECUTABLE |
               |         |
               |  TIC.EXE  |
```

Bibliographie

INSIDE OS/2, Gordon LetWin, Microsoft Press, 1988, 289 pages, ISBN 1-55615-117-9.
Ouvrage de présentation des objectifs et des principes d'OS/2. Explications des principes très claires. Un must pour celui qui désire connaître le fond des choses.
Programming the OS/2 Presentation Manager, Charles Petzold, Microsoft Press, 1989, 845 pages, ISBN 1-55615-170-5.
Ouvrage colossal, véritable cours de programmation de Presentation Manager. Très didactique et progressif. Accompagné d'un disque, nécessite de sérieuses notions de langage C. Indispensable pour le programmeur.
Microsoft Systems Journal, revue bimensuelle, Microsoft Corporation, ISSN 0889-9935.
C'est LA revue de Microsoft. Présentation en profondeur de sujets de très haut niveau relatifs aux produits Microsoft. Présentation claire, précise, accompagnée de diagrammes. Pas de publicité, 80 à 90 pages de technique. Listings disponibles par modem. Pour le programmeur de haut niveau.
OS/2 Programmer's Guide, Ed Iacobucci, McGraw-Hill,1988, 1100 pages, ISBN 0-07-881300-X.
Tour d'horizon d'OS/2 en mode « kernel». Description, usage et fonctionnement des appels système. Exemples en assembleur. Pour le programmeur familier des systèmes d'exploitation et de l'assembleur.
Software Tools for OS/2 : Creating Dynamic Link Libraries, Michel J. Young, Addison-Wesley, 1989, 343 pages, ISBN 0-201-51787-6.
Très bon ouvrage sur les librairies à chargement dynamique d'OS/2. Exemples en langage C. Très complet, aspect multitâche bien abordé.
The Programer's Essential OS/2 Handbook, David E. Cortesi, M&T Books, 1988, 709 pages, ISBN 0-934375-82-8.
Description du fonctionnement d'OS/2 en mode « kernel ». Appels système DOS, souris, clavier et écran. Dans le style « manuel de référence ». Accompagné d'un disque.
Microsoft OS/2 Programmer's Reference, Microsoft, 1989, 3 volumes de 740, 564 et 430 pages.
Trois volumes décrivant l'ensemble des appels système d'OS/2, tant en mode « kernel » que sous Presentation Manager. Le premier volume présente divers thèmes d'une façon assez générale mais néanmoins très précise. Les deux autres volumes sont les manuels de référence officiels de Microsoft. Indispensables au programmeur professionnel. Ces trois manuels font partie du kit de développement d'OS/2, version définitive.
Microsoft OS/2 Presentation Manager Softset, Microsoft, 1989, 197 pages.
Manuel d'utilisation des programmes faisant partie du kit de développement de Microsoft : LINK, BIND, IMPLIB, LIB, EXEHDR, MARKEXE, MKMSGF, MSGBIND, RC, DLGBOX,FONTEDIT, ICONEDIT. Impossible de programmer sérieusement OS/2 sans ces outils.

fl13



François Piette

Microsoft Quickhelp : écran obtenu en cliquant sur "OS/2 support" dans les options du compilateur

```
/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * *  * * * * * * * * */

Programme:      TIC.C
Auteur:         Franáois PIETTE
CrÇation:       Mai 1990
Objet:          Jeu de TIC TAC TOE,
dÇmonstration de programme simple
                mais complet en Presentation
Manager sous OS/2.
Compilateur:    Microsoft C 6.0

 * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * */
#define INCL_WIN
#define INCL_GPI
#include <os2.h>
#include <stdlib.h>
#include "tic.h"

#define DIVISIONS 3

MRESULT EXPENTRY ClientWndProc(HWND, USHORT,
MPARAM, MPARAM);
MRESULT EXPENTRY AboutDlgProc (HWND, USHORT,
MPARAM, MPARAM);


/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
int main(void)

{
    static CHAR  szClientClass[] = "TicTacToe";
    static ULONG flFrameFlags = FCF_TITLEBAR
| FCF_SYSMENU |
                                FCF_SIZEBORDER
| FCF_MINMAX  |

FCF_SHELLPOSITION | FCF_TASKLIST |
                                FCF_ICON
| FCF_MENU;
    HAB          hab;
    HMQ          hmq;
    HWND         hwndFrame, hwndClient;
    QMSG         qmsg;

    hab = WinInitialize(0);
    hmq = WinCreateMsgQueue(hab, 0);

    WinRegisterClass(hab, szClientClass,
ClientWndProc, CS_SIZEREDRAW, 0);

    hwndFrame = WinCreateStdWindow(HWND_DESKTOP,
WS_VISIBLE,

&flFrameFlags, szClientClass, NULL,
                                OL,
(HMODULE)NULL, ID_TIC,
                                &hwndClient);

    while (WinGetMsg(hab, &qmsg, NULL, 0, 0))
        WinDispatchMsg(hab, &qmsg);

    WinDestroyWindow(hwndFrame);
    WinDestroyMsgQueue(hmq);
    WinTerminate(hab);
    return(0);
}


/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
VOID DrawCross(HPS hps, LONG xLeft, LONG
yBottom, LONG xRight, LONG yTop)
{
    POINTL aptl[13];
    LONG    xSide, ySide;

    xSide = (xRight - xLeft  ) / 7;
    ySide = (yTop   - yBottom) / 7;

    aptl[ 0].x = aptl[11].x = aptl[12].x = xLeft
+ xSide;
    aptl[ 1].x = aptl[10].x = aptl[ 2].x = aptl[
9].x = xLeft + 3 * xSide;
    aptl[ 3].x = aptl[ 4].x = aptl[ 7].x = aptl[
8].x = xLeft + 4 * xSide;
    aptl[ 5].x = aptl[ 6].x = xLeft + 6 * xSide;

    aptl[ 8].y = aptl[ 9].y = yBottom + ySide;
    aptl[12].y = aptl[ 6].y = aptl[11].y =
aptl[7].y = aptl[10].y = yBottom + 3 * ySide;
    aptl[ 0].y = aptl[ 1].y = aptl[4].y = aptl[
5].y = yBottom + 4 * ySide;
    aptl[ 2].y = aptl[ 3].y = yBottom + 6 *
ySide;

    GpiSetColor(hps, CLR_BLUE);
    GpiBeginArea(hps, BA_BOUNDARY | BA_WINDING);
    GpiMove(hps, aptl);
    GpiPolyLine(hps, 12L, aptl + 1);
    GpiEndArea(hps);

}


/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
VOID DrawCircle(HPS hps, LONG xLeft, LONG
yBottom, LONG xRight, LONG yTop)
{
    ARCPARAMS arcp;
    POINTL    ptl;
    LONG      radius;
    LONG      xSide, ySide;

    xSide = xRight - xLeft;
    ySide = yTop   - yBottom;

    radius = min(xSide, ySide) / 3;

    arcp.lP = 1L; arcp.lQ = 1L;
    arcp.lR = 0L; arcp.lS = 0L;
    GpiSetArcParams(hps, &arcp);
    ptl.x = xLeft   + (xRight - xLeft  ) / 2;
    ptl.y = yBottom + (yTop   - yBottom) / 2;
    GpiSetColor(hps, CLR_RED);
    GpiMove(hps, &ptl);
    GpiFullArc(hps, DRO_OUTLINEFILL,
MAKEFIXED(radius, 0));
}

/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
MRESULT EXPENTRY AboutDlgProc(HWND hwnd, USHORT
msg, MPARAM mp1, MPARAM mp2)
{
    switch (msg) {
    case WM_COMMAND:
        switch (COMMANDMSG(&msg)->cmd) {
        case DID_OK:
        case DID_CANCEL:
            WinDismissDlg(hwnd, TRUE);
            return(0);

        break;
    }
    return(WinDefDlgProc(hwnd, msg, mp1, mp2));
}


/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
MRESULT EXPENTRY ClientWndProc(HWND hwnd, USHORT
msg, MPARAM mp1, MPARAM mp2)
{

    static HPOINTER hptrCircle , hptrCross,
hIcon;
    static SHORT
fBlockState[DIVISIONS][DIVISIONS];
    static SHORT xBlock, yBlock;
    static BOOL  fCrossState;
    HPS          hps;
    POINTL       ptl;
    RECTL        rcl;
    SHORT        x, y;
    HWND         hwndActionBar;

    switch (msg) {
    case WM_CREATE:
        hIcon      = WinLoadPointer(HWND_DESK-
TOP, (HMODULE)NULL, ID_TIC);
        hptrCircle  = WinLoadPointer(HWND_DESK-
TOP, (HMODULE)NULL, IDP_CIRCLE);
        hptrCross   = WinLoadPointer(HWND_DESK-
TOP, (HMODULE)NULL, IDP_CROSS);
        for (x = 0; x < DIVISIONS; x++)
            for (y = 0; y < DIVISIONS; y++)
                fBlockState[x][y] = -1;

        return(0);

    case WM_MOUSEMOVE:
        WinSetPointer(HWND_DESKTOP, fCrossState
? hptrCross : hptrCircle);
        return((MRESULT)1);

    case WM_SIZE:
        xBlock = SHORT1FROMMP(mp2) / DIVISIONS;
        yBlock = SHORT2FROMMP(mp2) / DIVISIONS;
        return(0);

    case WM_BUTTON1DOWN:
    case WM_BUTTON1DBLCLK:
        if (xBlock > 0 && yBlock > 0) {
            x = MOUSEMSG(&msg)->x / xBlock;
            y = MOUSEMSG(&msg)->y / yBlock;

            if (x < DIVISIONS && y < DIVISIONS)
{
                if (msg == WM_BUTTON1DBLCLK)
                    fBlockState[x][y] = -1;
                else {
                    fBlockState[x][y] =
fCrossState;
                    fCrossState        =
!fCrossState;
                }

                rcl.xRight = xBlock + (rcl.xLeft
= x * xBlock);
                rcl.yTop   = yBlock +
(rcl.yBottom = y * yBlock);
                WinInvalidateRect(hwnd, &rcl,
FALSE);
                WinSetPointer(HWND_DESKTOP,
fCrossState ? hptrCross : hptrCircle);
            }
            else {
                WinAlarm(HWND_DESKTOP,
WA_WARNING);
            }
        }
        else {
            WinAlarm(HWND_DESKTOP, WA_WARNING);
        }

        break;                          // do
default processing
```

```
    case WM_SETFOCUS:
        if (WinQuerySysValue(HWND_DESKTOP,
SV_MOUSEPRESENT) == 0)
            WinShowPointer(HWND_DESKTOP,
SHORT1FROMMP(mp2) ? TRUE : FALSE);
        return(0);

    case WM_CHAR:
        if (xBlock == 0 || yBlock == 0)
            break;

        if (CHARMSG(&msg)->fs & KC_KEYUP)
            break;

        if (!(CHARMSG(&msg)->fs &
KC_VIRTUALKEY))
            break;

        WinQueryPointerPos(HWND_DESKTOP, &ptl);
        WinMapWindowPoints(HWND_DESKTOP, hwnd,
&ptl, 1);

        x = max(0, min(DIVISIONS - 1,(SHORT)
ptl.x / xBlock));
        y = max(0, min(DIVISIONS - 1,(SHORT)
ptl.y / yBlock));

        switch (CHARMSG(&msg)->vkey) {
        case VK_LEFT:
            x—;
            break;
        case VK_RIGHT:
            x++;
            break;

        case VK_DOWN:
            y—;
            break;

        case VK_UP:
            y++;
            break;

        case VK_HOME:
            x = 0;
            y = DIVISIONS - 1;
            break;

        case VK_END:
            x = DIVISIONS - 1;
            y = 0;
            break;

        case VK_NEWLINE:
        case VK_ENTER:
        case VK_SPACE:
            WinSendMsg(hwnd, WM_BUTTON1DOWN,
                    MPFROM2SHORT(x * xBlock,
y * yBlock), NULL);
            break;

        default:
            return(0);
        }
        x =(x + DIVISIONS) % DIVISIONS;
        y =(y + DIVISIONS) % DIVISIONS;

        ptl.x = x * xBlock + xBlock / 2;
        ptl.y = y * yBlock + yBlock / 2;

        WinMapWindowPoints(hwnd, HWND_DESKTOP,
&ptl, 1);
        WinSetPointerPos(HWND_DESKTOP,(SHORT)
ptl.x,(SHORT) ptl.y);
        return(0);

    case WM_PAINT:
            hps = WinBeginPaint(hwnd, NULL, NULL);

            GpiErase(hps);

            if (xBlock > 0 && yBlock > 0) {
                for (x = 0; x < DIVISIONS; x++) {
                    for (y = 0; y < DIVISIONS; y++)
{
                        rcl.xRight = xBlock +
(rcl.xLeft  = x * xBlock);
                        rcl.yTop   = yBlock +
(rcl.yBottom = y * yBlock);

                        WinDrawBorder(hps, &rcl, 1,
1,
                                        CLR_NEUTRAL,
CLR_GREEN,
                                        DB_STANDARD |
DB_INTERIOR);

                        switch (fBlockState[x][y]) {
                        case 1:
                            DrawCross(hps,
rcl.xLeft,  rcl.yBottom,
                                        rcl.xRight,
rcl.yTop);
                            break;
                        case 0:
                            DrawCircle(hps,
rcl.xLeft,  rcl.yBottom,
                                        rcl.xRight,
rcl.yTop);
                            break;
                        }
                    }
                }
            WinEndPaint(hps);
            return(0);

        case WM_COMMAND:
            switch (COMMANDMSG(&msg)->cmd) {
            case IDM_ABOUT:
                WinDlgBox(HWND_DESKTOP, hwnd,
AboutDlgProc, (HMODULE)NULL,
                        ID_TIC, NULL);
                return(0);

            case IDM_CLEAR:
                for (x = 0; x < DIVISIONS; x++)
                    for (y = 0; y < DIVISIONS; y++)
                        fBlockState[x][y] = -1;
                WinQueryWindowRect(hwnd, &rcl);
                WinInvalidateRect(hwnd, &rcl,
FALSE);
                return(0);

            case IDM_QUIT:
                WinSendMsg(hwnd, WM_CLOSE, 0L, 0L);
                return(0);
            }
            break;

        case WM_DESTROY:
            WinDestroyPointer(hptrCircle);
            WinDestroyPointer(hptrCross);
            WinDestroyPointer(hIcon);
            return(0);
        }

    return(WinDefWindowProc(hwnd, msg, mp1,
mp2));

}

/* * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * */
```

```
;__________________
; TIC.DEF : fichier de dÇfinition de module
;__________________

NAME            TIC     WINDOWAPI

DESCRIPTION     'TIC-TAC-TOE (C) F. Piette,
1990'
PROTMODE
HEAPSIZE        1024
STACKSIZE       8192
EXPORTS         ClientWndProc
                AboutDlgProc

/*__________________
    TIC.H : dÇfinitions des constantes
__________________*/

#define ID_TIC          1
#define IDP_CIRCLE      2
#define IDP_CROSS       3

#define IDM_MENU        1
#define IDM_ABOUT       10
#define IDM_CLEAR       11
#define IDM_QUIT        12

/*
  TIC.RC : resource script file
__________________*/

#include <os2.h>
#include "tic.h"

POINTER ID_TIC          tic.ico
POINTER IDP_CIRCLE      ticcircl.ptr
POINTER IDP_CROSS       ticcross.ptr

MENU ID_TIC
{
  SUBMENU "~Menu",              IDM_MENU
    {
    MENUITEM "~A propos...",
IDM_ABOUT
    MENUITEM "~Efface",
IDM_CLEAR
    MENUITEM "~Quitte",         IDM_QUIT
    }
}

DLGTEMPLATE ID_TIC
{
    DIALOG "Tic Tac Toe", ID_TIC, 80, 100, 200,
80, FS_NOBYTEALIGN |
                FS_DLGBORDER | WS_VISIBLE |
WS_CLIPSIBLINGS | WS_SAVEBITS,
                FCF_TITLEBAR
    {
        ICON ID_TIC
-1, 90, 56,     0, 0
        CTEXT "DÇmonstration Presentation
Manager"    -1, 0, 38, 200, 8
        CTEXT "Copyright (C) Franáois PIETTE,
1990"   -1, 0, 28, 200, 8
        DEFPUSHBUTTON "Ok"
DID_OK, 80, 8, 40,  16, WS_GROUP
    }
}
```

OS/2 version 2.0

UN GRAND PAS EN AVANT

Le SDK d'OS/2 version 2.0 vient d'être annoncé par Microsoft. Il s'agit de la version d'OS/2 de 32 bits — réservée aux i386 et suivants — tant attendue.

Cette nouvelle version d'OS/2 marque une étape majeure dans l'évolution des systèmes d'exploitation pour les ordinateurs PC et compatibles. C'est la fin de la transition entre le mode réel de la famille des processeurs Intel 8086, et mode 32 bits des processeurs Intel i386 et i486.

OS/2 version 2.0 est un véritable système d'exploitation de 32 bits qui exploite au maximum les possibilités du i386. Il n'est plus question d'utiliser même un PC-AT traditionnel. Il faut au moins un 386sx (version en bus 16 bits) et mieux encore un 386dx, le nec plus ultra restant le i486. Rappelons que celui-ci combine dans un seul boîtier l'équivalent d'un processeur i386, d'un coprocesseur arithmétique i387, un gestionnaire de mémoire cache et une mémoire cache de 8 Ko. Le i486 peut donc être assimilé à un super i386, et nous n'évoquerons plus que ce dernier dans la suite de l'article.

La puissance du i386 est absolument nécessaire pour supporter des applications graphiques à hautes performances. La résolution des écrans et le nombre de couleurs ne cessent de croître, ce qui nécessite la manipulation d'un volume de données de plus en plus important.
Par le passé, on a fait travailler 50 ou 100 utilisateurs sur des systèmes bien moins performants qu'un i386, et cela sans sourciller. Par exemple, il y a dix ans, un mini-ordinateur de type VAX de milieu de gamme était moins puissant qu'un micro-ordinateur à base de i386 d'aujourd'hui ! Cela était possible car les applications d'alors fonctionnaient en mode caractère, sur des terminaux asynchrones lents. Aujourd'hui, personne ne serait prêt à abandonner l'interface graphique des PC ou du Macintosh sous prétexte que cela réclame un matériel plus puissant.

Le fait qu'OS/2 version 2.0 nécessite un 386 n'est pas étonnant. En 1988, dans son ouvrage intitulé « Inside OS/2 », Gordon Letwin, l'architecte d'OS/2, évoquait déjà la version spécifique 386 d'OS/2 « [..] OS/2 will support 32-bit protected mode applications that will run only on 80386 machines. »

La question reste posée de savoir si OS/2 saura s'imposer comme système d'exploitation utilisé sur la quasi-totalité des PC, ou s'il restera un système réservé aux stations de travail et aux utilisateurs de haut niveau. Personnellement, je penche pour la première solution.
On peut résumer les caractéristiques de version 2.0 par rapport à la version 1.2 de la manière suivante :

— système d'exploitation 32 bits pour processeurs 386 et suivants,
— totalement compatible avec OS/2 1.0, 1.1 et 1.2,
— support de MS-DOS grandement amélioré,
- modèle mémoire non segmenté (4 Go),
— mémoire virtuelle paginée, y compris pour le DOS,
— amélioration de l'interface graphique (GPI),
— amélioration de l'interface utilisateur,
— amélioration de la gestion des couleurs,
— intégration des polices TrueType (licence Apple !).

La liste est longue et pourtant il ne s'agit que d'un résumé. Nous allons passer en revue les points clefs de cette évolution.

compatibilité OS/2 version 1.x

La version 2.0 d'OS/2 accepte tous les programmes écrits pour OS/2 1.x sans aucune modification. La seule différence se situe au niveau des performances : la version 2.0 exécute les programmes plus rapidement.
La compatibilité avec DOS (MS-DOS ou PC-DOS) a été grandement améliorée. On peut dire que OS/2 version 2.0 offre un DOS plus performant que le DOS lui-même !

OS/2 travaille en mémoire virtuelle paginée (voir plus loin), y compris pour les applications DOS. Microsoft a intégré à OS/2 la technologie utilisée dans Windows 386, Windows 3, PC-MOS 386, Deskview (et d'autres encore...), à savoir celle des MVDM (ou Multiple Virtual Dos Machines). Plusieurs applications DOS peuvent ainsi fonctionner simultanément (et en multitâche).

L'architecture MVDM est basée sur une possibilité très importante du 386 : les machines virtuelles 8086 (mode V86). Dans ce mode, le i386 permet de gérer un nombre pratiquement illimité de machines virtuelles qui fonctionnent comme s'il y avait autant de processeurs 8086 distincts. Chacun de ces processeurs semblant fonctionner dans le mode réel habituellement employé par DOS.

Il y a deux intérêts majeurs à ces machines virtuelles: d'une part le code de gestion des machines et du DOS peut se trouver en fait totalement en dehors des fameux 640 Ko ; d'autre part, la gestion de la mémoire passe par le système de mémoire virtuelle. Il n'est donc pas nécessaire de disposer physiquement d'autant de fois 640 Ko qu'il existe de machines virtuelles DOS.

Avec l'architecture MVDM, les applications DOS tournent dans une fenêtre de Presentation Manager, ou en plein écran, ou en arrière-plan (multitâche) ou encore sous forme d'icône. Il reste plus de 600˘Ko disponibles sur les 640˘Ko (il s'agit là d'une des différences les plus visibles avec Windows 3.0). C'est mieux que toutes les autres versions de DOS de Microsoft (seul DR-DOS peut se targuer de résultats supérieurs !

Du point de vue OS/2, MVDM tourne comme une tâche ordinaire, profitant de la mémoire virtuelle paginée, du système de fichiers hautes performances (HPFS, introduit avec OS/2 1.2), et des facilités « couper-coller » de PM.
MVDM reste compatible avec l'émulation EMM (jusqu'à 8 Mo), l'XMS, le DOS 4.0 et les communications asynchrones jusqu'à 9600 bauds. Les machines virtuelles DOS peuvent être améliorées par des pilotes de périphériques virtuels (VDD, ou Virtual Device Driver).

Le miracle s'arrête là où commencent les « bidouillages » que se permettent certains programmes sous DOS :
— accès aux structures internes du DOS,
— accès directs au disque dur,
— cartes avec contrôleur DMA incorporé,
— conflits avec d'autres machines virtuelles.

modèle mémoire des applications en 32 bits
L'amélioration la plus significative de OS/2 n 2.0 par rapport à ses prédécesseurs est certainement l'adoption du système d'adressage linéaire de la mémoire (4 Go, ou 4096 Mo). Ceci correspond au modèle mémoire bien connu des programmeurs du 68000 et de la plupart des mini-ordinateurs.
La disparition du système de segmentation facilite le portage de programmes provenant d'autres environnements, telles les machines à processeurs RISC actuellement en vogue parmi les stations de travail pour ingénieurs.
Fondamentalement, la notion de segment n'a pas disparu. Le 386 permet de réserver 4 Go à un seul segment. Le modèle mémoire employé par OS/2 correspond donc à un seul segment de 4 Go. Dans ce segment, le programme d'application et ses données occupent les adresses faibles

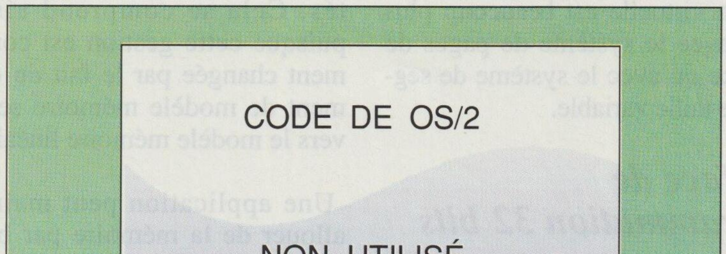


Figure 1 : modèle mémoire linéaire des applications 32 bits de OS/2 V2.0

(voir figure 1), tandis qu'OS/2 emploie les adresses hautes. Entre les deux, s'étendant vers le bas à partir de la frontière des 512 Mo se trouvent les données partagées ou allouées de manière dynamique. La disparition de la segmentation permet la gestion de données de taille arbitraire. Il n'y a plus de limite fixée à 64 Ko pour un objet déterminé. La gestion de la mémoire virtuelle est bien plus aisée avec le système de pagination : la mémoire physique est décomposée en pages de 4 Ko chacune.

Le gestionnaire de mémoire virtuelle se charge de déplacer les pages entre la mémoire centrale disponible et le disque dur. Il n'y a plus besoin de réorganiser les segments en mémoire centrale pour rassembler les petits morceaux de mémoire disparates. La gestion du fichier contenant la mémoire virtuelle est beaucoup plus simple avec le système de pages de taille fixe qu'avec le système de segments de taille variable.

interface de programmation 32 bits

Bien que les appels système 16 bits (API) aient été portés à 100 % pour assurer la compatibilité, les appels système 32 bits n'en sont pas une simple adaptation. Toutes les fonctions Vioxxx, Kbdxxx et Mouxxx qui servaient à la gestion de l'écran, du clavier et de la souris en mode caractère ont disparu des appels système 32 bits. L'intention des concepteurs de la version 32 bits est de forcer les programmeurs à utiliser les fonctions « DosRead » et « DosWrite » pour accéder à tous les périphériques, y compris la souris, le clavier et l'écran, ou alors d'utiliser les appels système de l'interface graphique Presentation Manager.

Cette intention est tout à fait louable si l'on songe aux dires de Microsoft lorsqu'il présente ses projets : porter OS/2 sur d'autres plates-formes que les PC. Dans ce contexte, il faut éliminer tous les appels système trop proches du matériel. Les programmeurs sont donc poussés à écrire soit de véritables applications 32 bits modernes utilisant Presentation Manager, soit à programmer de manière moins dépendante du matériel et donc plus portable dans le futur.

Mais que les fanatiques de la programmation typique des PC se rassurent : les appels système 16 bits des versions précédentes d'OS/2 existent toujours (compatibilité oblige) et peuvent être appelés par des applications 32 bits hybrides.

Les appels système relatifs à Presentation Manager sont restés pratiquement égaux à eux-mêmes, prouvant ainsi leur maturité.Un certain nombre d'appels système relatifs à la gestion de la mémoire ont été ajoutés. Cela se comprend aisément puisque cette gestion est complètement changée par le fait du changement de modèle mémoire segmenté vers le modèle mémoire linéaire.

Une application peut maintenant allouer de la mémoire par blocs de taille arbitraire. Ces blocs sont toujours arrondis à une taille multiple de 4 Ko, puisque c'est la taille des pages de la mémoire virtuelle. L'allocation s'effectue en deux temps : un premier pour réserver un bloc de mémoire de taille donnée, mais ne correspondant à rien ni physiquement, ni sur disque. Le second consiste à utiliser effectivement la mémoire et à créer son image virtuelle sur disque.

Ces deux temps compliquent un peu la tâche de programmation et obligent à connaître à l'avance la taille maximale qu'aura un objet. La segmentation avait l'avantage de pouvoir être extensible — dans les limites d'un segment — à volonté, puisque le système pouvait déplacer les segments. Dans la segmentation, seul le système connaît les adresses physiques réelles, tandis que dans l'espace linéaire, l'application connaît les adresses exactes. Les fonctions relatives à l'aspect multitâche ont été légèrement améliorées. Il y a plus de souplesse dans les possibilités de lancement de processus. On peut maintenant démarrer une nouvelle machine virtuelle avec un programme DOS. L'ordonnancement (scheduling) des tâches a été grandement amélioré. OS/2 2.0 est beaucoup plus rapide dans la recherche de la tâche devant avoir la main.

Les mécanismes de communication interprocessus par sémaphore ont été complètement revus dans la nouvelle version. Les anciens types de sémaphores ont été tous remplacés par de nouveaux, avec des appels système spécifiques à chaque type, évitant ainsi la confusion qui existait précédemment.

Finalement, le dernier point où des modifications essentielles ont été apportées est la gestion des exceptions (comme la division par zéro, l'accès illégal à la mémoire, les erreurs du coprocesseur, etc.) Il est maintenant permis d'enchaîner des gestionnaires d'exception, chacun pouvant décider s'il traite l'exception ou s'il la laisse à un gestionnaire situé plus loin dans la chaîne. OS/2 2.0 apporte donc une très grande amélioration dans les performances et les possibilités intrinsèques du système.

L'utilisateur retiendra d'une part un excellent support de multiples programmes DOS : le DOS sous OS/2 est meilleur que le DOS natif, et d'autre part, une interface graphique encore meilleure avec un bon support des palettes de couleurs.

Le programmeur retiendra qu'il ne devra pas modifier ses applications existantes grâce au support à 100 % des appels système 16 bits, mais qu'il aura du travail pour porter ses applications 16 bits non graphiques vers le monde 32 bits. Par contre, la migration des applications sous Presentation Manager 16 bits sera assez aisée.

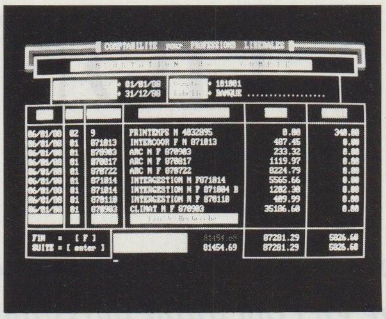
François Piette

PARTNER CONCEPT

22, rue des Panoyaux
75020 PARIS - Tél. : (1) 43 49 00 88

SI VOUS COMPTEZ SUR LA QUALITÉ... COMPTEZ SUR NOUS

SPÉCIALISTES EN GESTION ET COMPTABILITÉ, NOS LOGICIELS DEVIENDRONT VOS PARTENAIRES



PARTNER - 2035 - COMPTABILITÉ POUR PROFESSIONS LIBÉRALES

- Plan Comptable intégré, spécifiques aux professions libérales
- Saisie simplifiée des écritures et aides constantes
- Calculatrice sur écran à tout moment
- Mise à jour automatique, à chaque saisie, de la déclaration 2035

GESTION PARTNER GESTION COMMERCIALE ET FACTURATION

- Un seul et unique écran de saisie des Factures, Avoirs ou Proforma
- Création d'articles ou prestations pendant la saisie des factures
- Mise à jour automatique du journal des ventes et achats
- Mise à jour du stock en temps réel

COMPTA PARTNER COMPTABILITE GENERALE MULTI-SOCIETES

- Fenêtre d'affichage du Plan Comptable ou des journaux à la demande
- Sélection simple dans la fenêtre du compte ou des journaux à saisir
- Liste ou modification du brouillard en cours de saisie
- Compte de résultat en temps réel
- Consultation de l'historique d'un compte en cours de saisie

BON DE COMMANDE - DEMANDE DE DOCUMENTATION

Nom : ____________

Téléphone : ____________

Société : ____________

Adresse : ____________

Ville : ____________

Code postal : ____________

Signature : ____________

☐ Disquette 5 1/4 ☐ Disquette 3 1/2

☐	PARTNER -2035- Comptabilité Professions Libérales	1 990 F TTC
☐	Documentation	
☐	COMPTA PARTNER - Comptabilité multi-sociétés	3 700 F TTC
☐	Documentation	
☐	GESTION PARTNER Gestion Commerciale / Facturation	3 600 F TTC
☐	Documentation	
☐	PACK PROMOTION GESTION + COMPTA	5 400 F TTC
	Port	50 F TTC
	Ci-joint un chèque de	F TTC

Cerclez info pub 23

Calculateur HP— 48SX

LE PRÉSENT REJOINT LA SCIENCE-FICTION

Hewlett-Packard construit des calculateurs perfectionnés depuis une quinzaine d'année. Le dernier-né, la HP— 48SX marque un virage en conciliant puissance autonome et ouverture vers l'extérieur.

PREMIERE PARTIE : PRÉSENTATION

Jeudi 25 juin 1990, épreuve de mathématiques du baccalauréat, 7 h 47, dans le fond d'une salle du Centre d'examens d'Arcueil. Jean, le héros de notre histoire, découvre avec sérénité le sujet de mathématiques tant redouté. Le questionnaire porte la mention déjà classique « L'usage des calculatrices est AUTORISÉ ». Jean passe directement au problème sur 12 points.
La fonction à étudier est relativement simple f(x)=x*(ln(x+1)-ln(x). La HP—48SX que Jean a apportée enregistre la fonction grâce à une séquence de 20 touches, frappées au clavier. Le surveillant passe alors dans les rangs et vérifie le contenu de la mémoire vive des calculatrices. Arrivé à la table de Jean, l'examinateur sourit légèrement car il possède une HP—28S. Avec une maîtrise absolue, il réalise un « Memory Lost » et passe à la table suivante. Jean rit intérieurement. Ce que le surveillant ignore, c'est que la HP —48SX est dotée pour l'occasion de deux modules enfichables de mémoire vive, contenant toutes les informations de son cours de mathématiques, avec schémas et courbes associés, soit 288 Ko. Les deux « RAM Cards », configurées en mode Backup, restent totalement insensibles à toute remise à zéro. Jean retape la fonction. L'étude de celle-ci peut alors commencer. Le calcul de la dérivée ne pose aucun problème pour Jean. Après la frappe de 6 touches, la HP—48SX affiche f'(x)=ln(x+1)-ln(x)+x*(1/(x+1)-1/x). Il en va de même pour les extremums, les limites, et même certaines intégrales. Bien évidement, la

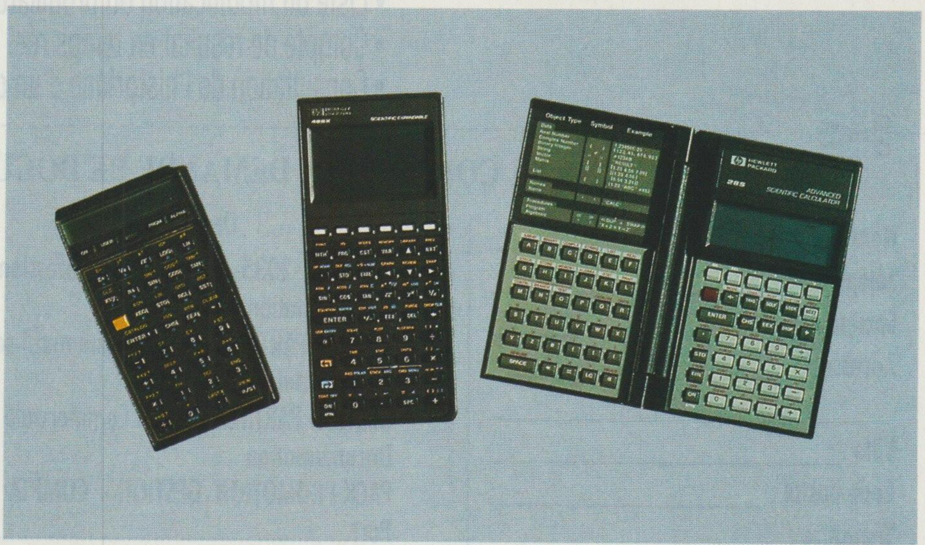
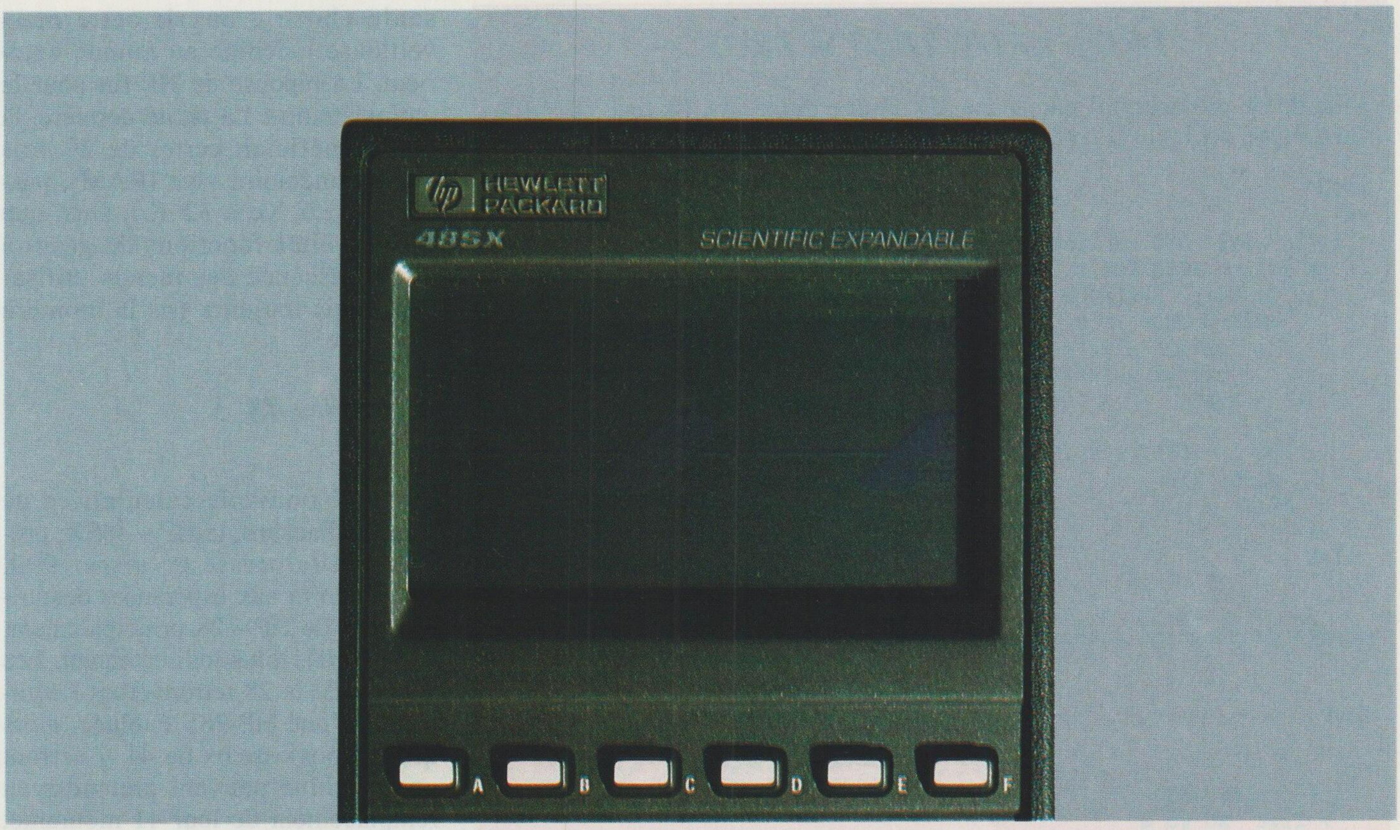


photo de famille : trois générations de HP X



exemple de tracé d'inéquation

HP—48 ne laisse pas tomber Jean dans cette épreuve difficile, elle lui donne (s'il le désire) toutes les étapes intermédiaires des calculs, et ce, avec une présentation semblable à celle de ses livres ! Le reste du problème ne posera pas de difficultés particulières à Jean, puisqu'il possède une multitude d'exercices type corrigés dans la mémoire de sa nouvelle HP. Grâce à sa HP—48SX, Jean quitte la salle d'examens l'esprit léger, il est assuré d'obtenir au moins 16/20.

retour vers le passé I

En 1979, Hewlett-Packard lança une calculatrice scientifique programmable qui fit sensation. Jugez plutôt : 64 registres de 7 octets chacun, soit 448 octets de mémoire vive, 20 Ko de mémoire morte, un écran à cristaux liquides à segments, capable d'afficher simultanément 12 caractères alphanumériques. Cette calculatrice, la HP}—41C, était programmable dans un langage rudimentaire très proche de la machine, et utilisait la notation polonaise inverse associée à une pile de taille limitée. Si la partie calcul de la machine était irréprochable (comme de coutume chez Hewlett-Packard), le génie des concepteurs de la HP—41 avait été de créer une machine ouverte au monde extérieur. La 41 disposait, dès sa première version, de quatre ports d'extension susceptibles de recevoir des modules de mémoire vive ou morte ou des périphériques, étendant ainsi les capacités de base de la machine en fonction des besoins spécifiques de l'utilisateur. Cette formule connût un tel succés auprès des étudiants et des scientifiques, que par la suite, la HP—41 fut déclinée au cours des années. A chaque nouvelle version, elle voyait sa capacité mémoire, ainsi que le nombre de ses fonctions augmenter. Le dernier modèle, la HP—41C X, est encore disponible aujourd'hui jusqu'à épuisement des stocks. Au cours des années 80, le nombre et la diversité des modules de mémoire, morte en particulier, n'ont cessé d'augmenter. Les domaines couverts sont aussi divers que la mécanique des fluides, l'aviation ou les finances.

retour vers le passé II

C'est en 1987 que Hewlett-Packard réalisa un nouveau coup d'éclat avec la sortie de la HP-28C. Ce calculateur provoqua une mini-révolution dans l'univers des calculatrices scolaires, dominé jusque-là par des marques japonaises du nom de Casio ou Sharp (respectivement 34 % et 21 % du nombre d'unités vendues en Europe, contre seulement 9 % pour HP).
Alors que la HP—41 était la première calculatrice alphanumérique digne de ce nom, la HP—28 devint la première calculatrice symbolique au monde. Dès son apparition, elle se vendit comme des « petits pains », et ce malgré un prix élevé pour ce type de matériel. Cette réussite, HP la devait à ses nouvelles capacités de calcul et de résolution symbolique. Rapidement, les utilisateurs de la 28 ne rêvèrent que d'une

PROJ, XROT, YROT et ZROT

Module de projection d'une scène 3D en une scène 2D. La liste en sortie est du type {(x1,y1) (x2,y2) ... (xn-1,yn-1) (xn,yn)}.

```
PROJ
<<
  1 {} 'slst' STO -> list xf yf zf
   << 1 list SIZE FOR i
       'list(i)' EVAL DUP 'MOV' if THEN
        ARRY-> DROP -> x y z
        << 'xf+(xf-x)*zf/(z-zf)' -> NUM
           'yf+(yf-y)*zf/(z-zf)' -> NUM
           R->C
        >>
       END 1 ->LIST slst SWAP +
       'slst' STO
       NEXT
   >>
 slst
>>
```

```
Module de rotation autours de l'axe OZ, de centre (xo,yo) et d'angle
a.

ZROT

<<
  1 {} 'slst' STO
  -> list  xo yo a
  << 1 list SIZE FOR I 'list(i)' EVAL
     DUP 'MOV' IF  THEN ARRY-> DROP
     -> x y z
     << 'xo+(x-xo)*cos(a)+(y-yo)*sin(a)' ->NUM
        'yo+(y-yo)*cos(a)-(x-xo)*sin(a)' ->NUM
         z 3 ->ARRY
     >>
    1 ->LIST slst SWAP +
    'slst' STO
    NEXT
  >>
 slst
>>
```

Les modules XROT et YROT sont identiques, exceptée la partie calcul que voici et le nom des variables locales contenant les valeurs des centres de rotation.

```
YROT

<<
 'xo+(x-xo)*cos(a)-(z-zo)*sin(a)' ->NUM
 y
 'zo+(z-zo)*cos(a)+(x-xo)*sin(a)' ->NUM
 3 ->ARRY
>>

XROT

<<
 'yo+(y-yo)*cos(a)+(z-zo)*sin(a)' ->NUM
 'zo+(z-zo)*cos(a)-(y-yo)*sin(a)' ->NUM
 x 3 ->ARRY
>>
```

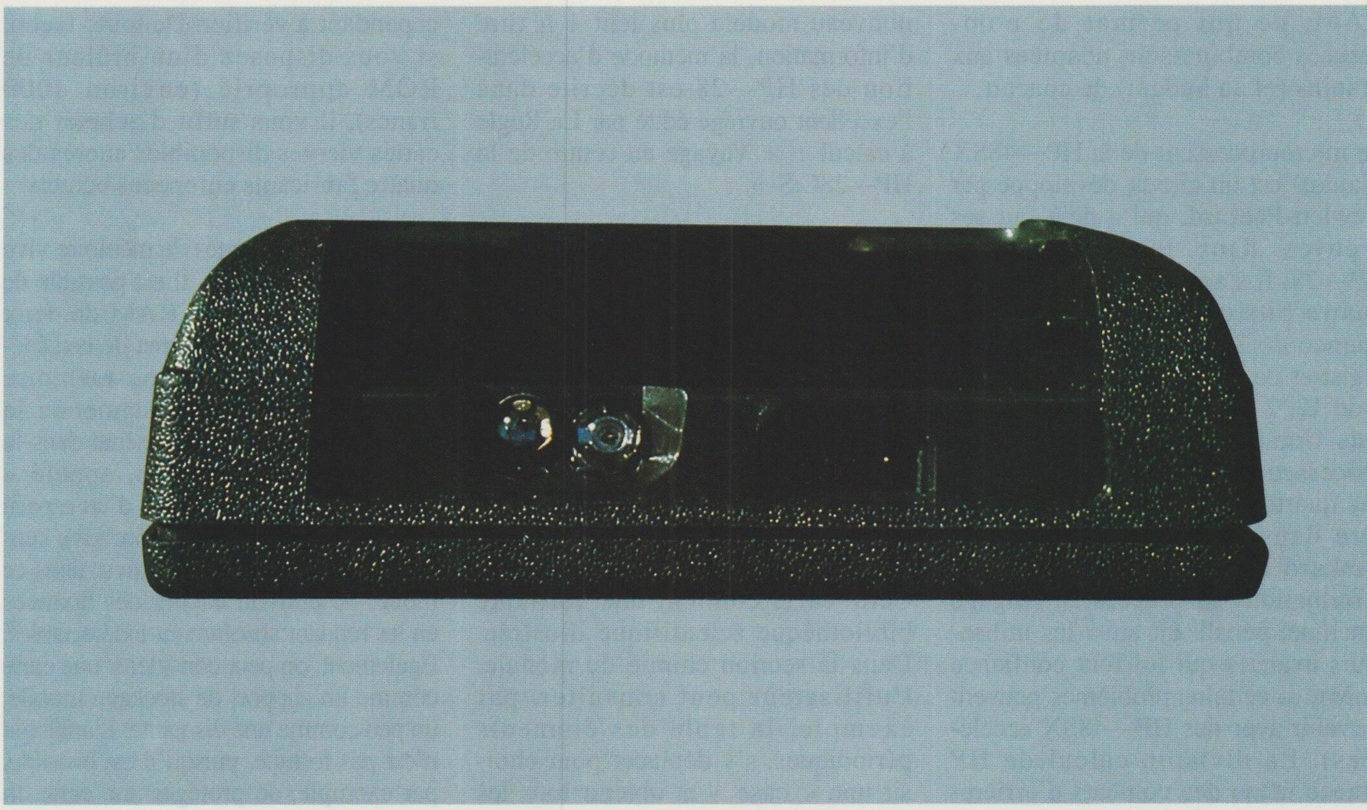
seule chose : ouvrir cette merveilleuse machine au monde extérieur. La réponse de HP fut pour le moins étrange. La petite dernière, la 28S, bénéficiait certes de 20 fois plus de mémoire vive (RAM), passant de 1,6 Ko à 32 Ko, ainsi que de nouvelles fonctions de gestion d'arborescence des menus utilisateur, mais toujours pas la moindre ouverture !

41 + 28 = 48 !

La toute nouvelle calculatrice de Hewlett-Packard, la HP—48SX, présentée à la presse en mars 1990, répond enfin aux espérances des utilisateurs de HP—28 principalement, et de HP041 plus indirectement. Les adeptes de la 28 retrouveront l'équivalent d'une HP-28S évoluée, alors que les possesseurs de 41 y verront une machine pouvant prétendre au remplacement de leur 41 mourante, tout en gardant le même esprit.
Pour Hewlett-Packard, la première cible visée est celle des professionnels, et plus particulièrement les ingénieurs et techniciens. HP espère donc renouveler un parc de quelque 350 000 HP—41 en milieu professionnel pour le seul continent européen. Bien entendu, les 5 000 000 de techniciens en Europe qui ne connaissent pas encore les calculateurs HP pourront également se laisser séduire. La seconde cible de la 48SX, est constituée par le monde des étudiants, plus précisément celui des élèves ingénieurs qui représente près de 1 000 000 de personnes en Europe. Bien évidement, la 48SX est autorisée aux examens en France, et ce sont donc globalement tous les étudiants —fortunés— qui sont visés, ainsi que leurs professeurs.

DEUXIEME PARTIE : LE MATÉRIEL

Le premier contact physique avec la 48 se passe généralement bien. La machine conserve des dimensions raisonnables : 18,03 cm x 8,13 cm x



la connectique de la HP-48SX : entrée/sortie infra-rouge, port série, deux connecteurs pour carte mémoire.

2,92 cm. Avec ses 255 grammes, elle tient bien en main et peut encore se loger dans certaines poches. L'orientation générale de la machine est verticale comme sur la 41 et reste dans la ligne des calculatrices Hewlett-Packard. Les utilisateurs de 28 seront un peu déroutés au début, mais on s'habitue à tout : adieu double clavier mobile et orientation horizontale.
Au niveau du clavier, précisons tout de suite un petit problème. La 48 offre plus de 2100 fonctions accessibles par seulement 49 touches. Le résultat est un clavier surchargé, qui reste toutefois exploitable avec de l'habitude. L'utilisateur peut toutefois reconfigurer ce clavier à volonté (des caches pour le clavier sont vendus en option).

l'écran et le clavier bien conçus

L'écran est surprenant par sa taille et sa qualité pour une machine de cette catégorie. Sans évoquer le chemin parcouru depuis la HP-41 (comparons ce qui est comparable), les détenteurs de 28 découvriront avec joie un afficheur à cristaux liquides « super-twisted » de 8 lignes de 22 caractères, soit le double de celui de la 28. Cet écran est constitué en fait d'une matrice graphique de 131 x 64 pixels. Le contraste peut être réglé pour s'adapter à toutes les conditions de travail. Mais un écran sans caractère n'est rien. Avec la 48, vous pouvez afficher les 255 caractères ascii de la machine directement à l'écran, alors que c'était impossible avec la 28 qui ne pouvait que les imprimer. Ce jeu de caractère est très riche. On retrouve l'intégralité des caractères accentués des langues européennes. Il est même possible d'accéder aux caractères accentués utilisés en hongrois ! Pour en finir avec les caractères, signalons la présence des principaux symboles mathématiques et physiques, ainsi que la possibilité d'avoir certains caractères en différentes tailles. On pourrait presque parler de polices de caractères... Enfin presque.

la mémoire et le processeur 4 bits

Un des points les plus significatifs de l'évolution de la HP—48SX par rapport à la HP—28S est la mémoire. La HP—48SX, dans sa version de base dispose de 256 Ko de mé-moire morte (ROM) et, comme la 28S, de 32 Ko de mémoire vive. Il est intéressant de signaler aux utilisateurs de HP-28, qu'une grande partie de la ROM de cette dernière semble avoir été conservée sur la 48, simplement étendue pour les nouvelles fonctions. Sur la face arrière, une trappe donne accès à deux logements d'extensions qui offrent la possibilité d'étendre la mémoire de la machine. Les modules enfichables ont une taille comparable à celle des cartes de crédit et sont un peu plus épais. Il existe deux capacités pour ces cartes : 32 Ko ou 128 Ko, aussi bien en mémoire vive que morte. La HP—48 peut donc être étendue à un maximum de 288 Ko de mémoire vive (32 Ko de base plus deux fois 128 Ko en modules). La mémoire morte (ROM) peut, elle, atteindre 512 Ko (256 Ko de base plus deux modules de 128 Ko). Il est possible de panacher les carte ROM et

RAM, ce qui permet de nombreuses combinaisons adaptées aux besoins (et au budget) de chacun.

Le microprocesseur de la HP—48SX (Saturn) est un circuit développé par Hewlett-Packard, qui a déjà fait ses preuves dans les HP—71 ou HP—28. Il s'agit d'un processeur de quatre bits qui travaille à la fréquence de 2 MHz. Il dispose de 9 registres de 64 bits, 3 de 20, 2 de 16, 1 de 12, 2 de 4 et 1 bit pour la retenue. Une particularité de ce microprocesseur est qu'il inverse l'ordre des quartets à la lecture ou à l'écriture dans la mémoire. Hewlett-Packard France bloque toutes les informations sur l'accès au langage machine, pénalisant ainsi les utilisateurs avancés qui lui font confiance (même si certains problèmes peuvent survenir avec des HP—48SX accélérées). La division calcul de HP France mène des combats d'arrière-garde, disparus depuis longtemps dans le monde des PC par exemple. La rétention d'information n'a jamais été une attitude payante avec la presse, sans doute une optique à mettre au compte du manque de concurrence dans ce secteur. Nous avons essayé les méthodes déjà éprouvées avec la HP—28S, sans pour autant obtenir de résultat, excepté des plantages suivis d'un « Memory Lost »» ! Les possesseurs de HP—48SX intéressés par ce type de manipulation auront intérêt à se mettre en rapport avec le club des utilisateur HP, qui a plus de chance de posséder ces informations. L'attitude de HP France est d'autant plus regrettable que l'information technique sur les calculateurs circule plus difficilement en France qu'aux Etats-Unis et que la 48SX s'avère assez lente. En effet, même si la fréquence d'horloge du SARTURN de la 48 est plus élevée que celui de la 28, et que les tests démontrent une vitesse supérieure à celle d'une HP—28S de base, le résultat global en vitesse d'exécution reste très inférieur à celui de la 28S accélérée. Difficile de convaincre un possesseur de 28S accélérée d'échanger sa machine contre un nouveau modèle plus lent ! A titre d'information, la méthode d'accélération des HP—28 est décrite dans l'excellent ouvrage édité par La Règle à calcul : « Voyage au centre de la HP—28C/S ».

l'utilisation des cartes ROM et RAM

Une fois enfichée, une carte de mémoire morte (ROM) apporte des fonctions supplémentaires aux 2100 fonctions initiales. Pour le moment, une seule carte est disponible, le module de référence pour la résolution de problèmes : la HP-SOLVE. Cette carte contient une véritable bibliothèque scientifique illustrée. Dans la section chimie du module, l'utilisateur peut consulter, par exemple, la table des éléments périodiques, s'y déplacer pour choisir une « case » et obtenir tous les renseignements utiles sur cet élément, ainsi que les graphes. La section physique traite aussi bien de la mécanique des fluides que de la thermophysique. Les principales formules y sont présentes avec les schémas associés ! Pour les développeurs qui voudraient se lancer dans la mise en carte de logiciels, il semble que pour la France, ce soient les distributeurs qui s'occuperont de l'opération et non pas Hewlett-Packard. Cette information reste cependant à vérifier. De toute façon, si vous disposez d'un brûleur de ROM approprié (environ 4000 francs), il vous suffit d'acheter des cartes vierges disponibles auprès des quatre fabricants européens actuels.

L'utilisation des cartes de mémoire vive (RAM) est différente. Il est possible de configurer une carte RAM de deux façons. La première permet de considérer la carte comme une mémoire d'extension qui vient augmenter la mémoire de base déjà présente dans la machine. L'autre mode, appelé « Backup », dote la machine d'un espace de sauvegarde de la mémoire vive centrale. Il est normalement prévu, dans ce mode, de pouvoir cacher des données en les rendant absolument indétectables. Egalement, on peut considérer une carte comme un support de stockage mobile, un peu comme une disquette. L'analogie n'est pas fortuite, puisqu'il est possible, par exemple, de protéger une carte en écriture. La mémoire de la carte, une fois retirée de la machine, est sauvegardée par une pile bouton assurant une autonomie de près de trois ans. Si la batterie au lithium d'une des cartes venait à s'épuiser, un message « LowBat(Px) » s'afficherait (où « x » désigne le numéro de la carte en péril) tant que l'utilisateur ne remédierait pas au problème.gEnfin, en plus des logements d'extensions, la HP—48SX possède une interface série et une liaison bidirectionnelle à infrarouge.

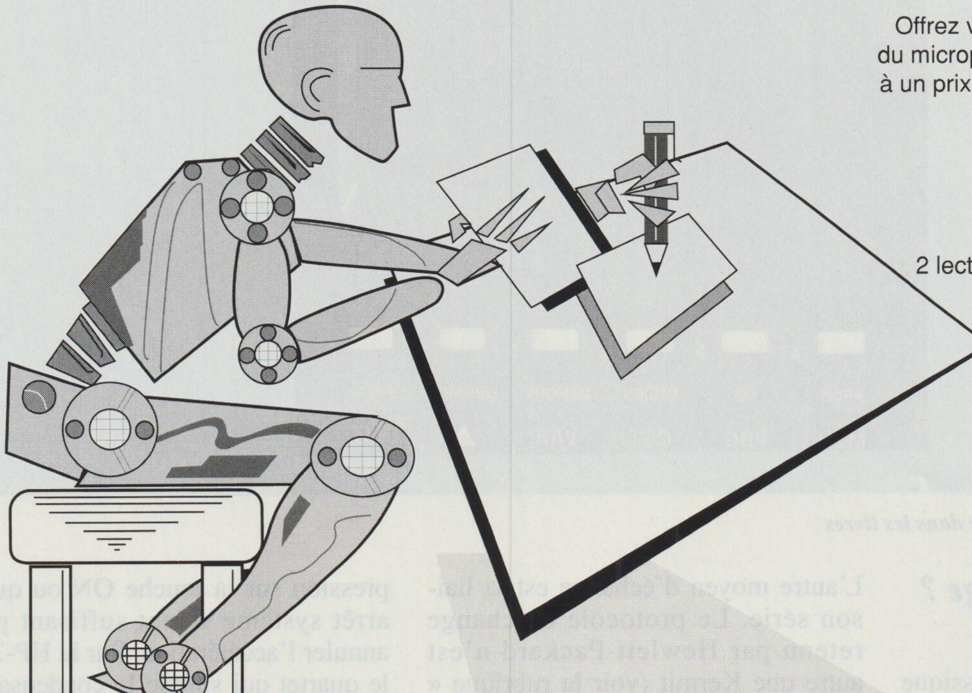
tracé d'une équation paramétrée



POUR AFFRONTER LES ANNEES 90

MV TECH 386 SX

Offrez vous la puissance
du microprocesseur 386
à un prix MICRO VIDEO !!!



1 Mo extensible 8 Mo
Carte controleur
2 lecteurs disquettes / 2 disques durs
1 lecteur 5' 1/4 1.2Mo
ou 3'1/2 1.44Mo
2 séries / 1 parallèle
Disque dur 20 Mo / 28ms
Clavier 102 touches
Ecran monochrome
haute résolution 720 x 348
Ms-Dos 4.01

8423 F.HT

(9990 F TTC)

Toutes options possibles:
Disque dur fixes ou amovibles
Ecrans couleur / multisync

PROMO du mois (Uniquement à Paris)
ATARI PC2 8088 / 512K / D Dur 30 Mo / Carte EGA
4990 F TTC

La disponibilité peut varier suivant les magasins

Dans la limite des stocks disponibles

Prix différents en Belgique, renseignez vous !

MICRO VIDEO

la passion d'un spécialiste, la puissance d'une chaine.

PARIS
8, rue de Valenciennes 75010 Paris
☎ 40.37.92.75 / 40.34.97.80 +
Ouvert du Mardi au Samedi de 10H à 19 H
Métro: Gare de l'Est / Gare du Nord

BORDEAUX
3, cours Alsace et Lorraine
33000 Bordeaux
☎ 56.44.47.70

DAX
56, Avenue Victor Hugo
40100 Dax
☎ 58.74.18.63

LYON
11,12 cours Aristide Briand
69300 Caluire
☎ 72.27.14.74

NOUVEAU !
MONTPELLIER
7 rue Raoux
34000 Montpellier
☎ 67.58.39.20

NOUVEAU !
NANTES
6 rue Mazagran
44000 Nantes
☎ 40 69 15 92

PERPIGNAN
8, avenue de Grande Bretagne
66000 Perpignan
☎ 68 34 24 40

TOURS
81, rue Michelet
37000 Tours
☎ 47.05.78.50

BELGIQUE

BRUXELLES
1, rue Dons
1050 Bruxelles
☎ 02 / 648 9074

DINANT
21 place communale
5198 ANHEE
☎ 082 / 611451



Equation Writer, les formules comme dans les livres

rouge, vous avez dit rouge ?

La sortie infrarouge est un classique chez Hewlett-Packard. Alors que sur la HP—28, cette liaison servait uniquement à communiquer avec l'imprimante de la même marque, avec la 48, on découvre une nouvelle dimension dans la communication sans fil, puisque la liaison est bidirectionnelle et que deux HP—48 peuvent échanger directement toute information. Que les lycéens se calment tout de suite, il n'est pas possible d'effectuer des transferts sur de grandes distances. Le manuel recommande une distance maximale de l'ordre de 5 cm, il nous a cependant été possible de monter jusqu'à 10 cm. La raison de cette limitation à la réception des données est évidente : quel ministère de l'éducation autoriserait l'utilisation d'une calculatrice permettant l'échange d'informations sans contrôle possible ? L'émission de données vers l'imprimante est, pour sa part, possible sur près d'un mètre (avis aux bricoleurs).

L'autre moyen d'échange est la liaison série. Le protocole d'échange retenu par Hewlett-Packard n'est autre que Kermit (voir la rubrique « Qu'est-ce que... » d'Action PC n° 4). Ceci permet de connecter n'importe quel appareil disposant d'une interface RS-232. Il peut s'agir d'un PC ou compatible, d'un Macintosh, voir d'un Atari ST (quitte à écrire le « driver » au lieu d'attendre que Hewlett-Packard le fasse). L'interface série de la HP— 48 SX utilise seulement quatre fils : transmission, réception, signal et masse. La tension du signal est de 3,5 volts. Ce sont là les rares informations disponibles actuellement.

Que ce soit par la liaison infrarouge ou par la sortie série, les vitesses de transfert sont de 1200, 2400, 4800 ou 9600 bauds. Reste à signaler que dans l'hypothèse d'une accélération, des problèmes surviendraient certainement pendant les transmissions de fichiers. Espérons cependant que l'effet du SPEED soit, comme sur la HP-28S, éphémère, et qu'une simple pression sur la touche ON ou qu'un arrêt système soient suffisant pour annuler l'accélération. Sur la HP-28S, le quartet qui simule le condensateur de la HP—28C était remis à l'état « normal » dans ces mêmes conditions et la machine repassait alors en vitesse normale (c'est-à-dire lente).

Les périphériques

Hewlett-Packard destine la 48 au remplacement de la HP-41. Comment amener les plus farouches adeptes de la 41 à cette nouvelle machine ? La réponse semble évidente : commercialiser un émulateur de HP—41 au format carte d'extension ! Hélas, cette carte n'était pas disponible au moment de notre test. Outre les cartes de mémoire et l'incontournable imprimante à liaison infrarouge, Hewlett-Packard propose deux kits d'interface série pour connecter la calculatrice respectivement à un PC ou à un Macintosh. C'est bien entendu la version PC qui a retenu notre attention. Ce kit paraît

indispensable (ainsi que le PC qui va avec) pour exploiter la HP—48SX . N'oublions pas que la 48 ne possède pas de lecteur de carte magnétique (contrairement à la HP—41). Avec le kit PC, il devient possible de sauvegarder l'intégralité ou une partie de la mémoire utilisateur de la HP—48 sur disquette ou disque dur, rien n'empêche de créer des copies de la RAM adaptées à une tâche précise (maths, physique, chimie, comptabilité, etc.), puis de les recharger dans la machine en cas de besoin. Il est même possible avec le kit d'utiliser la HP—48 à partir d'un PC comme une sorte de supercoprocesseur mathématique (lent... très lent par rapport aux produits PC !).

HP se commet avec le PC

Pour réaliser cet essai, nous avons eu la chance de disposer d'un des rares câbles de liaison disponible sur le territoire français. Il nous fut malheureusement impossible de nous procurer les logiciels utilitaires normalement fournis avec ce câble. Parmi ceux-ci, se trouve par exemple un logiciel de conversion d'objet GROB (Graphic Object) en fichier TIFF, un chronomètre, un programme CAT qui permet de visualiser l'ensemble des fonctions intégrées de la machine, avec les différents arguments nécessaires comme sur les HP—28. Y figure également un logiciel permettant d'exploiter le « scheduler » de la HP—48SX sur votre PC. En dépit du des logiciels, le câble de liaison a retenu notre attention.

Pour faire fonctionner l'ensemble HP—48SX et PC, outre l'indispensable câble, vous devez disposer sur votre PC d'un protocole Kermit. Une fois le logiciel lancé, il faut configurer le PC en 9600 bauds, grâce à un habile SET BAUD 9600 et un SET PARITY NONE. Concernant la HP—48SX, il est impératif de régler les paramètres d'entrée/sortie au moment de la première utilisation. Par la suite, il suffit de sauvegarder la liste IOPAR (Input Output Parameter) sous un nom particulier (PCPAR par exemple) et de la renommer en IOPAR lors de l'utilisation suivante. Voici le réglage utilisé pour le test :

programme principal ANIM

```
<< CLLCD                                  Effacer l'écran
   -> scripte                             Enregistre le script dans
                                          une variable locale
                                          du même nom
   << 1                                   Met 1 dans le premier
                                          niveau de la pile
      scripte                             Met une copie du
                                          script dans la pile
      SIZE                                Détermine la taille
                                          du script
      3 /                                 Divise cette taille par 3
                                          pour avoir le nombre
                                          d'ordres à exécuter
      FOR I                               Début d'une boucle sur I
                                          de
                                          1 au nombre calculé
      scripte
      I
      I 2 +
      GET                                 Prend la sous-liste du
                                          script comprise
                                          entre I et I + 2
      LIST->                              Transforme la sous-liste
                                          en éléments de la pile +la
                                          taille de la sous-liste
      DROP                                Jette la taille de la sous
                                          liste
      EVAL                                Evalue le nom de la
                                          commande du script
      NEXT                                Passe à l'ordre suivant
                                          dans le script
      >>
   -> result
      << 'result(1)' EVAL 2               Trace l'image à l'écran
      result  SIZE FOR I                  en utilisant les 'MOV'
      result I GET LIST-> DROP            pour lever le "stylos"
      DUP 'MOV' IF SAME THEN
      DROP2 'result(I+1)' EVAL
      DUP END DUP ROT SWAP
      LINE
      NEXT DROP
      >>
   LCD-> OFF                              Sauve l'écran et éteint
    >>
    G8
```

IR/wire wire
ASCII/binary ASCII
baud 9600
parity none 0
checksum type 1
translate code 1

Une fois ces formalités réglées, et le

Le calcul symbolique

Pour la majorité des gens, une calculatrice est juste bonne à donner des résultats numériques. Ce point de vue était justifié jusqu'à l'apparition de la HP—28 qui offrait des possibilités de calcul symbolique. Prenons un exemple pour expliciter ce dont il s'agit. Affectons deux valeurs numériques (5 et 2) à deux variables A et B. Avec une calculatrice conventionnelle, le résultat de l'expression « (A+B) 2 » donne 49. En utilisant les capacités symboliques de la HP-48SX, l'utilisateur n'est plus obligé de donner une valeur numérique à une variable pour obtenir un résultat. Dans notre exemple, si l'utilisateur demande à la HP—48 de développer l'expression « (A+B) 2 », le résultat est « A 2 +2.A.B+B 2 ». Examinons un autre exemple trivial. Quelles sont les solutions de l'équation « 3.X 2-4.X+5=0 ». Une calculatrice normale précisera que l'équation ne possède pas de solution réelle et peut-être donnera-t-elle les solutions complexes. La HP—48 donne deux solutions complexes, à savoir « 2/3+-(166761/150841)*i ». En résolution symbolique, elle répond « X=(4+s1*(0+20800333/313622*i))/6 » (« s1 » est une variable qui prend pour valeur « -1 » et « 1 »). Laissons tomber les exemples simplistes, et passons à des équations plus conséquentes. Soit « cos(X^ln(tan(X)))=sqrt(X) ».

La dérivation de cette équation par rapport à X donne « (1/180*(1/180*(1+tan(X)^2)/tan(X)*ln(X)*X^ln(tan(X))*pi+ln(tan(X))*X^(1+ln(tan(X))))*sin(X^ln(tan(X)))*pi)=1/2/sqrt(X) ». Ouf ! Bref, le calcul symbolique c'est ça. La machine travaille et raisonne comme on le ferait sur une feuille de papier, sauf que peu d'entre nous peuvent trouver la solution.

object story

Cette notion de calcul symbolique cache en fait un concept qui fait toute la puissance des calculatrices HP-28 et 48. Il s'agit de la notion d'objet. La machine traite les données stockées dans sa mémoire en fonction de leur catégorie. Il existe sur la HP—48, 18 types d'objets. Chaque objet est soumis à certaines régles et possède certaines propriétés. Le calcul symbolique ne fait qu'exploiter les objet algébriques délimités par des «'».

Voici la liste des objets disponibles sur la HP—48SX.

Liste des objets communs à la HP—48SX et HP—28 :

nombre réel
nombre complexe
entier binaire
matrice
nom
objet algébrique — Outre la possibilité de calcul symbolique, cet objet permet à ceux qui ne peuvent se faire à la notationpolonaise inversée de travailler en notation algébrique.
programme
chaîne
liste

Liste des nouveaux types d'objets de la HP—48SX :

objet graphique (Pour la HP—28, une image était une chaîne).
objet “marqué” — Ce nouvel objet est n'importe lequel des 18 objets, à la différence près que l'utilisateur peut lui affecter un nom ou un identificateur. Ce nouveau type est très utile pour reconnaître la nature d'un élément dans la pile.
unité
répertoire
sauvegarde
librairie
nom de librairie
fonction intégrée
commande intégrée

Ajoutons à cela un ensemble d'objets système que la machine utilise pour son bon fonctionnement. Pour le moment, nous n'en savons pas plus sur ces objets.

port série de la HP ouvert (OPENI), le travail peut commencer. On est loin ici des bricolages de la HP 28 S qui ne fonctionnaient que très aléatoirement après des heures de tâtonnements. Envoyer un fichier de la HP—48SX vers le PC est un jeu d'enfant. Il suffit pour cela de se positionner dans le menu où se trouve l'objet en question, de placer son nom dans le premier niveau de la pile, de taper R NomDeFichier sur le PC, et, enfin, de frapper la commande SEND sur la HP—48SX dans le menu I/O. Après quelques secondes de patience, le disque dur du PC permet d'apprécier le résultat ! L'opération en sens inverse est tout aussi simple. Vous pouvez par exemple saisir votre programme avec votre éditeur préféré sur votre PC (au diable le clavier de la HP), puis l'envoyer vers la HP—48SX. Celle-ci se charge de convertir le fichier ASCII en binaire pendant la transmission. Le paramètre TRANSIO (Translate Code) prend ici toute son importance car des caractères (code 128 à 159 et code 160 à 255) de la HP—48SX ne peuvent être affichés sur certains modèles d'ordinateurs. Les caractères dont le code est compris entre 160 et 255 sont pris en compte par la norme ISO 8859. Les autres nécessitent un logiciel adéquat. Pour remédier à ce problème, il

suffit de régler judicieusement le paramètre TRANSIO :

0 pas de translation du code des caractères non éditables,
1 conversion du 10 CHR en CHR 13 (retour chariot HP ->PC),
2 translation des codes compris entre 128 et 159,
3 translation de tous les codes non éditables.

Pour piloter la HP grâce au PC, ou vice versa, il suffit d'utiliser le mode SERVER de Kermit. Les principales fonctions de Kermit sont disponibles, et leur utilisation permet une gestion optimale de la mémoire de la HP—48SX. Un remède qui marque la fin des cauchemars nés des « MEMORY LOST », et autres « OUT OF MEMORY ».

D'autres périphériques sont annoncés par Hewlett-Packard comme une carte de liaison avec un rétro-projecteur pour utiliser la calculatrice dans le cadre d'un cours de mathématiques par exemple.

Gestion de la mémoire utilisateur

Afin de résoudre le problème de disproportion entre le nombre de touches du clavier et le nombre de fonctions disponibles, Hewlett-Packard a adopté la solution suivante : utiliser six touches de fonction juste sous l'écran. La dernière ligne de l'écran permet de connaître à chaque instant la fonction de ces six touches grâce aux cinq premières lettres du nom de l'objet affichées à l'écran en vidéo inversé. (Une petite nouveauté par rapport à la HP—28 : les touches portant le nom d'un répertoire ou sous-répertoire sont surmontées d'une discrète rangée de pixels.) Ce système permet à l'utilisateur de naviguer dans un ensemble de menus et sous-menus. En ce qui concerne la mémoire vive utilisateur, la solution est analogue. On peut créer dans une véritable arborescence. C'est exactement comme avec un disque dur. Par rapport à la HP-28S, on peut rappeler un répertoire et ses répertoires fils dans la pile, puis le recopier ailleurs (c'est ce qui explique la présence de l'objet répertoire). Une autre nouveauté vient du fait qu'il est possible grâce aux commandes « ATTAC » et « DETAC » de rattacher et détacher une librairie d'un module à un niveau de l'arborescence. Pour faciliter la navigation dans les menus utilisateur, en plus de la commande « HOME » (le « cd \ » du DOS), la fonction UP permet de remonter aux répertoires pères (fonction équivalente au « cd.. » du DOS).

une tortue qui se prendrait pour un lièvre

Le seul gros problème de la machine vient de sa vitesse (voir le passage ci-dessus sur le microprocesseur et le tableau de mesures). Bien que cette lenteur puisse s'expliquer vue la complexité des algorithmes mis en jeu, pourquoi diable ne pas fournir le moyen d'accélérer la machine, au moins pour les utilisateurs avancés ?

Test de vitesse

Lors de la réalisation de ces tests, les programmes et leurs données sont strictement identiques d'une machine à une autre, excepté le mode de lecture du timer interne qui nécessite un SYSEVAL sur la HP—28S. Plus le nombre obtenu est petit, plus la vitesse est grande. nombres aléatoires et sommation de ceux-ci avec contrôle de fin de pile.

```
<< 1 1000 START RAND NEXT
  DO IFERR + THEN KILL END
  UNTIL KEY END

<<
```

test 2 : affichage des mille premiers entiers à l'écran.

```
<< 1 1000 FOR I I 1 DISP NEXT
>>
```

test 3 : inversion d'une matrice 10 x 10.

```
<< 1 100 START RAND NEXT
  {10 10} ->ARRY INV
```

test 4 : créer une table des COS et SIN de 0 à 365.

```
<< 1 365 FOR I
  I COS I SIN NEXT
  {2 365} ->ARRY
>>
```

test 5 : création de 1000 sous répertoires.

```
<< 1 1000 START 'A' CRDIR A
NEXT >>
```

test 6 : tri d'une liste de 50 nombres aléatoires.

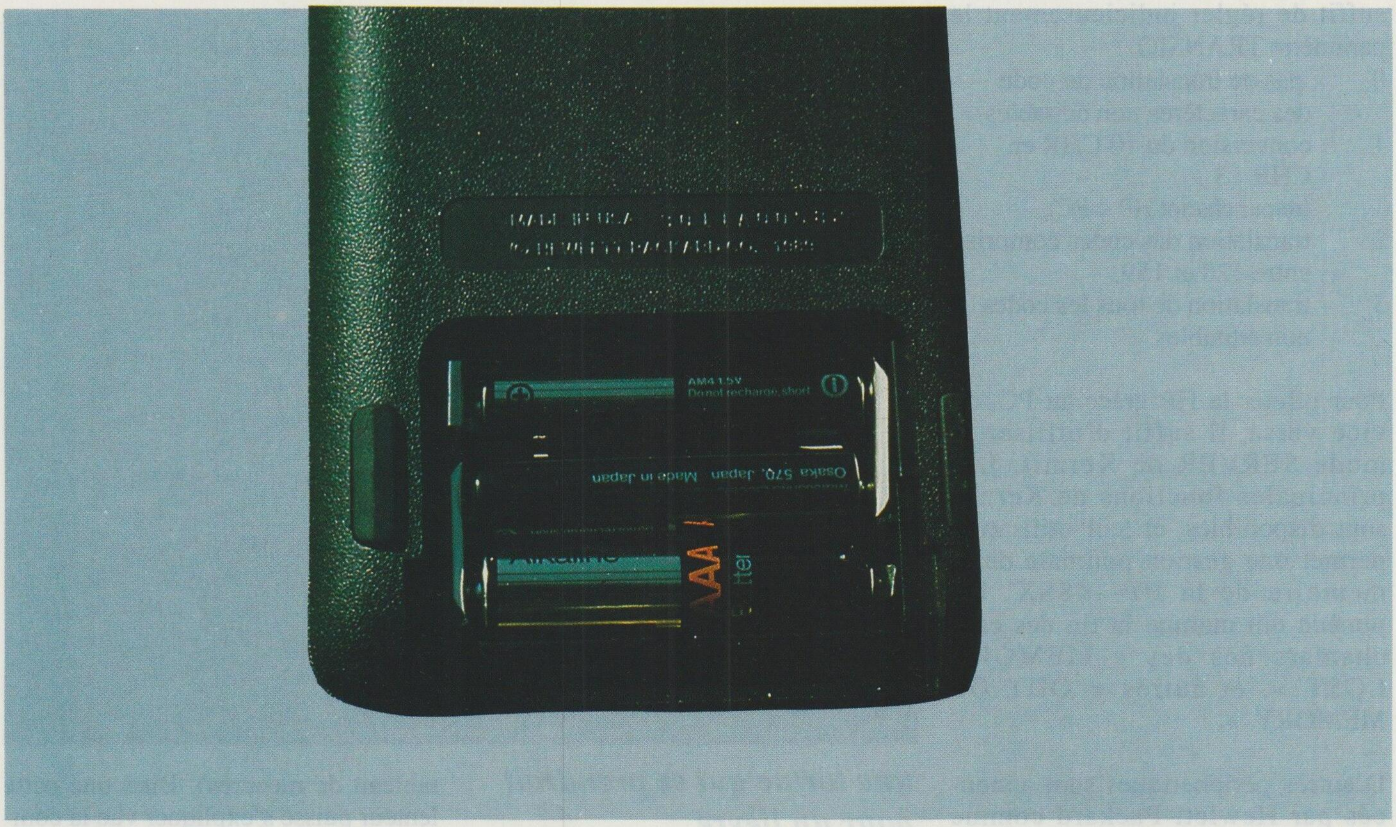
```
>>
<< 1 50 START RAND NEXT 50 -
>LIST

 DUP 'LST' STO SIZE O 1 ->
TAIL TEMP P

 << 1 TAIL 1 - FOR I I 'P' STO
    I 1 + TAIL FOR J 'LST(J)'
EVAL

 'LST(P)' EVAL IF < THEN J 'P'
STO
    END NEXT 'LST(P)' EVAL
'TEMP' STO

 'LST(I)' EVAL 'LST(P)' STO
   TEMP 'LST(I)' STO NEXT
```



des piles R6, tout simplement !

Du côté de chez Hewlett-Packard, la raison est simple, l'opération est impossible. Evoquant le logiciel SPEED commercialisé aux USA chez tous les bons revendeurs HP, on nous a, dans un premier temps, répondu que c'était une fausse rumeur !
Depuis, nous savons avec certitude que ce logiciel existe bel et bien outre-Atlantique. L'accélération était (déjà) impossible sur les HP—28, pourtant aujourd'hui la 28C (intervention au niveau du matériel donc annulant la garantie), comme la 28S (voir ci-dessus) acceptent de travailler plus vite. Qu'en est-il pour la HP— 48 ? Il semblerait que la méthode à utiliser soit similaire mais identique à celle employée sur la HP—28S.

TROISIEME PARTIE : LES LOGICIELS

HP Equation Writer

C'est peut-être l'application qui servira le plus aux élèves nuls en maths ! Ce nouvel outil permet de saisir directement les équations les plus complexes comme elles sont imprimées dans les manuels. Si la taille de l'équation venait à dépasser celle de l'écran, celui-ci se transforme alors en une fenêtre mobile. Le calculateur vérifie en permanence la syntaxe de l'équation et signale si nécessaire toute écriture contre nature. Cette débauche de moyens se paye en précieuses secondes, et une fois l'effet de nouveauté passé, il est plus rapide de saisir l'équation sans l'Equation Writer. Jugez-en vous-même : la saisie de l'équation « sin(w1*t+sin(w2*t+p)) » nécessite 18 secondes en mode directe et 30 secondes avec l'Equation Writer. Dommage, car une version plus rapide serait devenue un classique du genre.

HP Solve

Cette application déjà présente sur les HP—28, a vu son ergonomie améliorée. Il est possible à présent de balayer l'ensemble des variables d'une équation avec leur valeur et unité. Pour les non-initiés, cette application permet d'obtenir, entre autres choses, des prévisions à partir d'une équation en modifiant les variables, ou de déterminer la valeur d'une inconnue.

Prenons l'exemple saisi précédemment avec l'Equation Writer. Dans HP Solve, affectons aux variables w1, w2 et p les valeurs 10, 10,5, et 0,01. Après 5 secondes de calcul, la HP signale un changement de signe à « t=-4.87803934346 E 4 ». Certaines fonction très utiles sont regroupées dans un menu particulier, vous pouvez par exemple déterminer une aire, des extremums, des racines, des dérivées ou des intersections de courbes sans pour autant quitter le graphe des yeux.
Le luxe est même poussé jusqu'à afficher les coordonnées du curseur. La dernière ligne de l'écran sert à afficher l'action des touches de menu. Si l'espace manque pour apprécier pleinement le dessin de la courbe, un scrolling vertical est à la disposition de l'utilisateur !

Graphismes

Avec l'arrivée des dernières calculatrices japonaises, le graphisme est rapidement devenu indispensable sur un calculateur. Avec la HP—48, il est possible de réaliser des graphes dans huit modes différents : Par rapport à la HP—28, les courbes se dessinent plus vites, elles sont continues, même avec des pentes quasi verticales. Une nouvelle fonction d'auto-ajustement des paramètres de tracé a été ajoutée. Mais la section graphisme ne s'arrête pas aux seules courbes mathématiques, il est possible d'ajouter des éléments graphiques sur une courbe, comme des ellipses, des boîtes ou des lignes, allumer ou éteindre un pixel.

Mode	exemple type d'équation
équation	y = sin(x)
conique	2x - 3y + 3z - 5 = 0
polaire	4 sin(teta) = r
paramétrique	t - t + i(t^3 - 3t)
inéquation	(cos(x) <= 1/2) or (x = 1/4)
barre	statistiques
histogramme	statistiques
nuage de points	statistiques

	HP-28S	HP-28S accélérée	HP-48SX
test 1	30 s	18 s	25 s
test 2	43 s	23 s	39 s
test 3	10 s	10 s	10 s
test 4	32 s	16 s	23 s
test 5	6 mn 24 s	3 mn 46 s	4 mn 38 s
test 6	5 mn 17 s	2 mn 40 s	3 mn 46 s

la gestion des unités

Quel étudiant peut affirmer ne jamais avoir eu de problèmes avec les unités ? Avec la HP—28, il était déjà possible de gérer la quasi-totalité des unités rencontrées dans le monde. Avec la HP—48, ce sont 148 unités regroupées en 16 catégories qui sont disponibles. Les catégories sont longueurs, surfaces, volumes, temps, vitesse, masses, forces, énergies, pressions, électricité, angles, lumière, radiations et viscosités. Il est possible de mélanger toutes ces unités pour obtenir par exemple des « m^2.Kg.A^2/V » ou des « "mileUS".min.°R ».

En fins connaisseurs que vous êtes, vous aurez sans doute remarqué que cette dernière unité est non réglementaire (SI). Pas de problème, une simple pression sur la touche adéquate et on obtient des « 86333282,344.m^2.s.K ». Cette nouvelle fonction remplace astucieusement le catalogue des données dimensionnelles des HP—28. Dans le même style, il est possible de réaliser des factorisations sur les unités.

l'agenda

Ce petit utilitaire peut s'avérer utile pour ceux qui oublient souvent leurs rendez-vous. Une gestion complète du temps est possible comme avec la plupart des « scheduler » du marché. Ce petit plus, même s'il n'est pas in-dispensable à une calculatrice scientifique, se révèle utile dans la rédaction de program-mes utilisant le temps comme donnée principale. Il n'est plus nécessaire de passer par des « SYSEVAL » intempestifs et dangereux. En parlant de SYSEVAL, cette instruction meurtrière mais ô combien utile sur les HP—28, il faut signaler qu'il est désormais possible de tenter un sauvetage de la mémoire vive (RAM) principale en cas d'accident.

Un dernier mot sur les utilitaire, le gestionnaire de pile est fantastique. Sur les HP-28 et 48, la taille de la pile n'est limitée que par la taille de la RAM. Il arrive donc quelquefois (bien que ce ne soit pas recommandé pour la vitesse d'exécution des programmes et des commandes) d'avoir plus d'une centaine de niveaux de pile. Avec ce nouvel outil, il est tout à fait possible de se déplacer dans cette pile avec les touches du curseur et de travailler sur celle-ci avec les touches de fonctions !

les mathématiques

Les mathématiques constituent le domaine de prédilection des calculatrices Hewlett-Packard. La HP—48 est, à ma connaissance, le plus puissant calculateur de poche. Il couvre tous les domaines du calcul numérique avec une précision de 12 chiffres, dans l'intervalle [-9,99999999999*104E99,-1*10E499] U {0} U [1* 10E499, 9,99999999999*10499]. La HP—48SX ne se contente pas des seuls réels, mais traite avec la même aisance les nombres complexes, les entiers binaires et d'autres bases usuelles. Les fonctions sont nombreuses et diverses.

l'algèbre

Les fonctions algébriques de la HP—28 se retrouvent intégralement sur le nouveau modèle. Il est tou-

jours possible d'obtenir des dérivées symboliques de toutes les fonctions imaginables, de réaliser des factorisations et des développements et, grande nouveauté, il est enfin possible de faire calculer des intégrales symboliques de certaines équations. Je décèle déjà une étrange lueur dans les yeux de certains étudiants. Qu'ils se rassurent, la HP—48 ne peut intégrer que des fonctions simples, combinaisons linéaires des fonctions intégrées. Par exemple « 1/(1+x″) » donne après intégration sur [0,1] le résultat suivant (atan(x)/dx(x)|x=1-atan(x)/dx(x)|x=0) ». Si un résultat symbolique est impossible à trouver, il reste toujours la possibilité de faire un calcul approché.

les matrices

Une déception quant au calcul matriciel. Certes la HP—48SX effectue tous les calculs possibles, comme les inversion ou les élévations à une puissance. La déception vient du fait qu'il est toujours impossible d'intégrer des objets symboliques dans les matrices.

Il reste cependant la possibilité de travailler avec des matrice complexes. La nouveauté se situe dans la saisie desdites matrice. En effet, elle s'opère comme avec un tableur où chaque cellule représente un élément de la matrice. L'édition s'en trouve grandement simplifiée et contrairement à l'Equation Writer, elle est pratique et rapide. La présentation de toutes les fonctions mathématiques de la HP—48 est impossible. Les deux manuels d'utilisation ne font pas moins de 851 pages ! Signalons en bref des fonctions remarquables comme le calcul vectoriel en 2D ou 3D, l'isolation d'une variable symbolique, calcul des solutions d'une équation du second degré, ou de n équations à n inconnues, sommations (le produit n'est pas présent), calcul des approximations de Taylor ou factoriser des équations en mettant les nombres réels sous forme fractionnaire !

la programmation

Hewlett-Packard a doté sa dernière calculatrice du langage de programmation des HP—28. Ce langage est le RPL (Reverse Polish Lisp). En fait, ce langage est un subtil mélange de Lisp pour le traitement des listes, de Forth pour la création d'instructions utilisateur, et de Pascal pour les structures de contrôle évoluées comme le « REPEAT / UNTIL » ou le « WHILE ». Pour certains, ce langage reste totalement incompréhensible et illisible, alors que pour d'autres, il est très agréable, souple et pratique. Rappelons qu'avec la sortie de l'émulateur de 41, la HP—48 dispose de deux langages (plus le langage machine), et qu'il est question d'en développer un troisième proche du Basic, destiné à tourner à la fois sur HP—48 et PC.

Signalons aussi côté PC, le projet d'un émulateur complet de HP—48SX qui permettrait de réaliser tous les programmes sur le PC avant de les charger, une fois terminés, sur le 48. Mais pour le moment, il faut se contenter du RPL et du langage machine (non encore utilisable). Pour présenter rapidement le RPL, étudions pas à pas le programme principal d'un ensemble de modules qui assurent la représentation d'objets en 3D modifiés suivant un petit script.

Les objets constituant la scène à animer sont regroupés dans une même liste. Dans cette liste, chaque objet est séparé par un nom global 'MOV'. Chaque objet est décrit par les sommets de chaque arête sous forme de vecteur.

Exemple : la scène {[-1 1 0][1 1 0][1 -1 0][-1 1 0][-1 1 0] MOV [1 1 0][1.5 2 3][2 1-1][1 1 0]} représente un carré plan etun triangle plan incliné.Le script {0 0 20 XROT 0 0 -15 YROT 0 0 35 ZROT 0 0 -10 PROJ}effectue une rotation de 20° autour de l'axe OX de centre (0,0) suivie d'une rotation de centre identique autour de l'axe OY, etc. Le script doit impérativement se terminer par une projection sur le plan de l'écran.

Cet exemple n'a pas la prétention de remplacer le manuel de programmation (le manuel de programmation avancée est en option), mais seulement de présenter l'esprit général du RPL.

Les modifications apportées au langage initial de la HP—28 sont peu nombreuses, mais fort agréables. Le test « CASE » fait son apparition, ainsi que le « INPUT » qui faisait cruellement défaut à la 28. Le débogueur de la HP—28 a été légèrement amélioré et reste toujours aussi utile et agréable à utiliser.

acheter ou ne pas acheter ?

Nous sommes en présence d'une machine très intéressante qui peut prétendre à la succession de la HP—41 et qui apporte de nombreuses améliorations par rapport à la HP—28.

L'ouverture de la HP—48SX laisse présager d'un rapide développement des modules d'extension, ainsi que des périphériques. Outre le prix élevé de la machine, sa vitesse globale très inférieure à celle de la HP—28S accélérée reste le facteur négatif pour l'achat de la machine par un étudiant. Quant aux techniciens, à eux de voir si la vitesse est primordiale ou non.

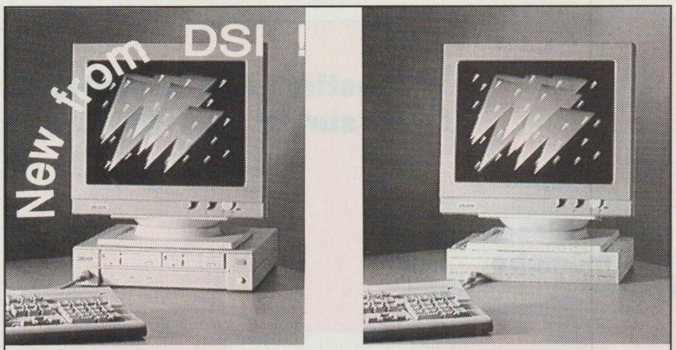
Jamel Tayeb

286 7990 TTC **386SX 10990 TTC** **386-20 15990 TTC**

Avec 1Mo Ram, Lecteur disquette, Disque dur 20Mo 38ms, Ecran. Garantie 1 An.



Tarif complet et devis pour toute configuration spécifique sur simple demande.
Disque dur de 20 à 710Mo. Carte d'extension. Imprimante. Conseil. Logiciel. Formation.



Big Power Is Getting Small !

AT 286 12Mhz LAN STATION.
Disque dur optionnel 20 à 200Mo. Ecran Monochrome ou Couleur Compatible NOVELL. Deux emplacements disponibles pour cartes Poste de travail de petite dimension pour bureau secretariat, poste de direction et étudiants !

A partir de 6990F TTC

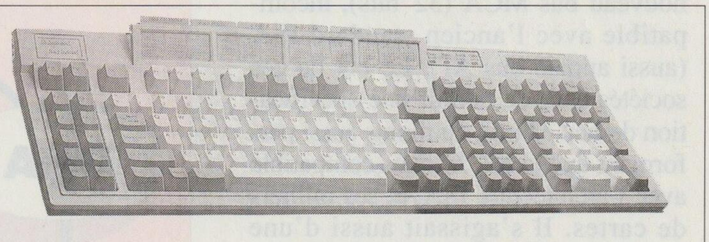
Avec écran hercules et un lecteur de disquette 1.44Mo !



Track Ball **590F**

Souris **390F**

Scanner 400dpi **1590F**

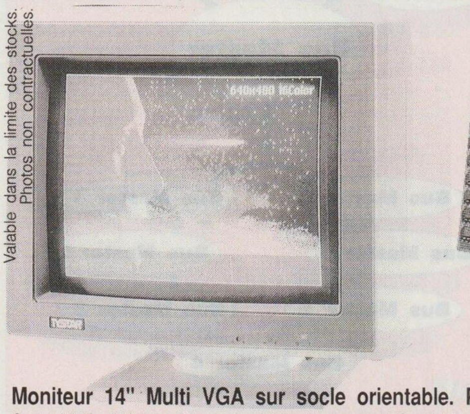


Exclusif !

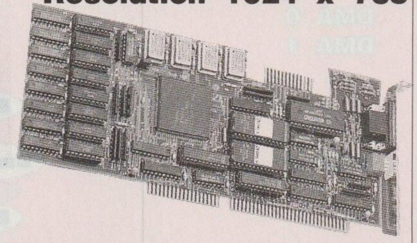
La Calculatrice en Plus !

Clavier 130 touches avec calculatrice intégrée dans le pavé numérique. Ecran LCD 8 caractéres. Compatible IBM PC et PS/2 (option)

Valable dans la limite des stocks. Photos non contractuelles.



KIT VGA Couleur Plus
Carte + Moniteur 14"
Résolution 1024 x 768



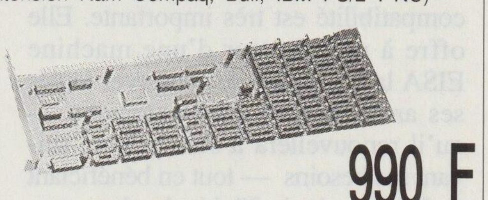
Moniteur 14" Multi VGA sur socle orientable. Pitch 0.28
Carte Vidéo VGA multimodes (CGA, Hercules, EGA, VGA, 800x600 et 1024*768 (en option). Jusqu'à 256 couleurs simultanées.

3990* F TTC

* Supplément sur un système DSI Monochrome. L'ensemble seul 5490 TTC

CARTE MEMOIRE

Supporte de 1 à 6Mo. Mode Etendu et EMS 4.0
(Extension Ram Compaq, Bull, IBM PS/2 : NC)



990 F

Le Mo de Ram : 1000 F TTC — Equipée de 0Ko

DSI STRASBOURG

16 Rue du Fossé des Treizes
67000 STRASBOURG
TEL 88.75.15.89 - 88.32.57.02

Cerclez info pub 25

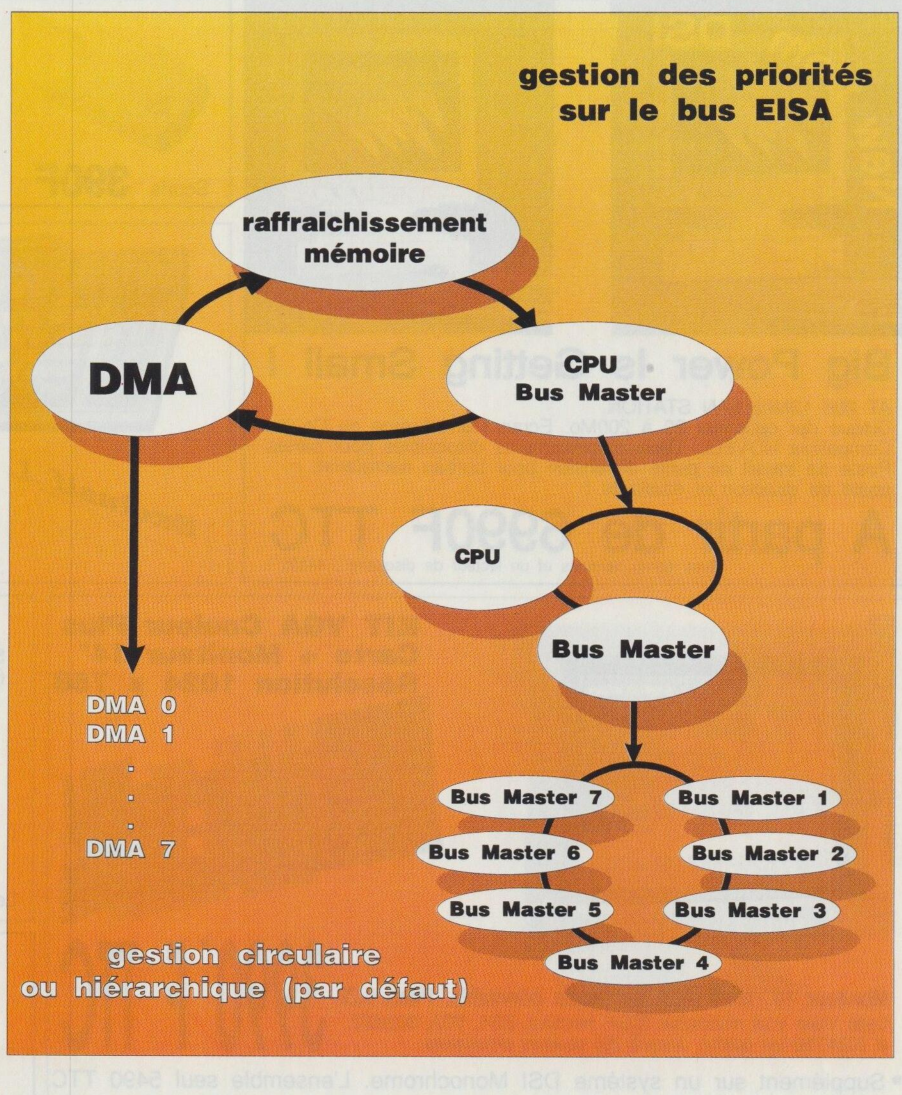
REVENDEUR CONSULTER DAVEL TRADING AU 88.32.20.08 !

High Tech...

LE BUS EISA

Depuis quelques mois, le bus de 32 bits EISA réussit une timide percée sur le marché des PC. Ses qualités et sa compatibilité en font indéniablement le standard de demain.

Le bus EISA ou, en d'autres termes, le bus ISA étendu a été conçu par un consortium de neuf entreprises (AST Research, Compaq, Epson, Hewlett-Packard, NEC, Olivetti, Tandy, Wyse et Zenith). Ce consortium fut créé suite à l'annonce par IBM des PS/2 et de leur nouveau bus MCA (32 bits), incompatible avec l'ancien standard ISA (aussi appelé bus AT). Le but de ces sociétés était de définir une spécification de bus 32 bits, au moins aussi performant que le MCA, mais compatible avec l'architecture ISA, et ses milliers de cartes. Il s'agissait aussi d'une réponse à IBM qui réclamait des royalties sur tous les éventuels compatibles PC à bus MCA. Le bus EISA est le résultat de deux ans de travail du consortium. Il reste conforme à la philosophie d'ouverture et de compatibilité ascendante du PC. La notion de compatibilité est très importante. Elle offre à un acheteur d'une machine EISA la possibilité de conserver toutes ses anciennes cartes d'extension — qu'il renouvellera à son rythme, suivant ses besoins — tout en bénéficiant de la technologie 32 bits la plus avancée si nécessaire.

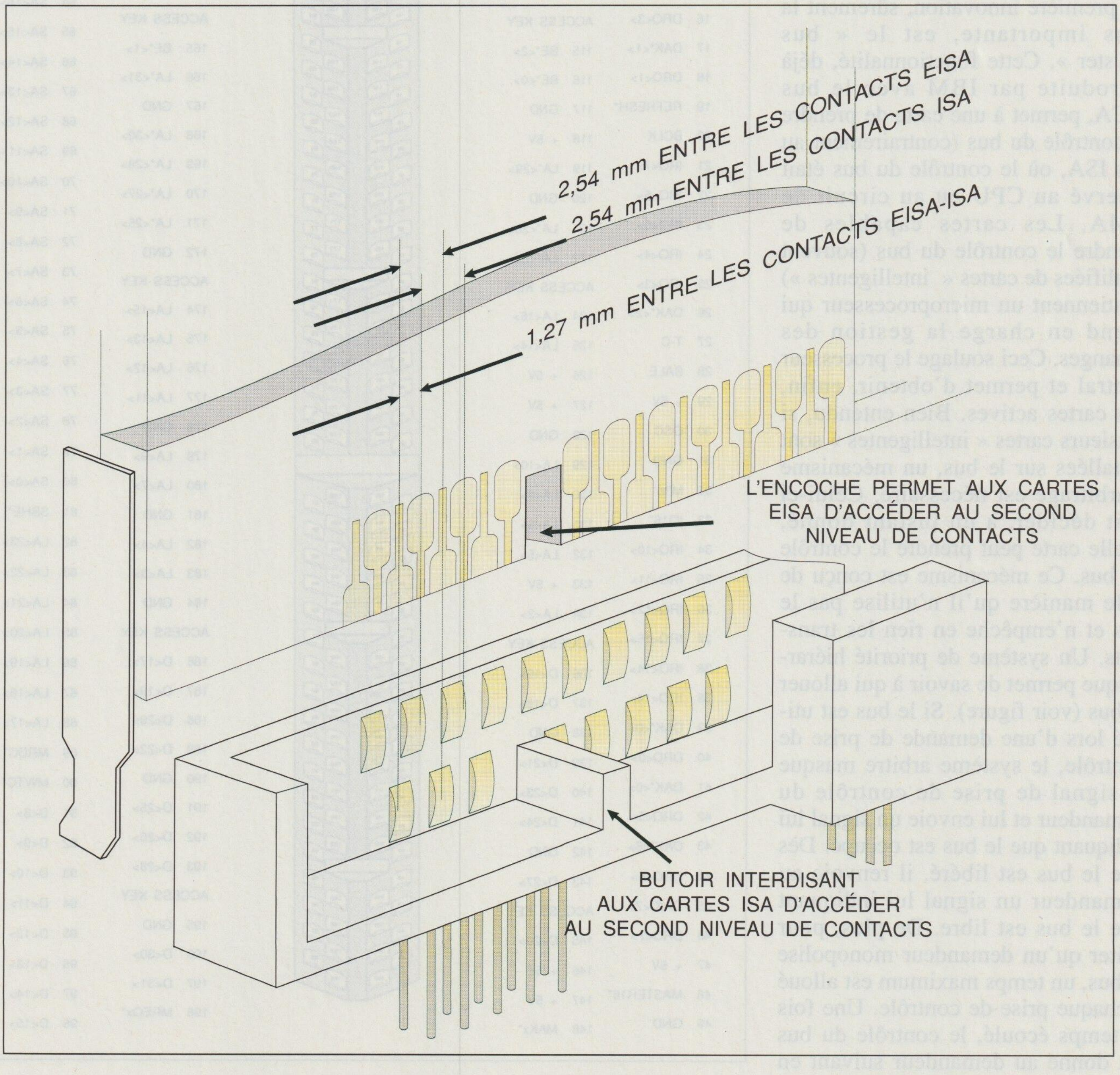


Avant d'examiner le bus EISA (Extended Industry Standard Architecture), précisons ce qu'est le

bus ISA (Industry Standard Architecture). Du point de vue fonctionnel, c'est sur ce bus que sont placés les connecteurs où viendront s'enficher les cartes d'extensions.

les bus PC et AT

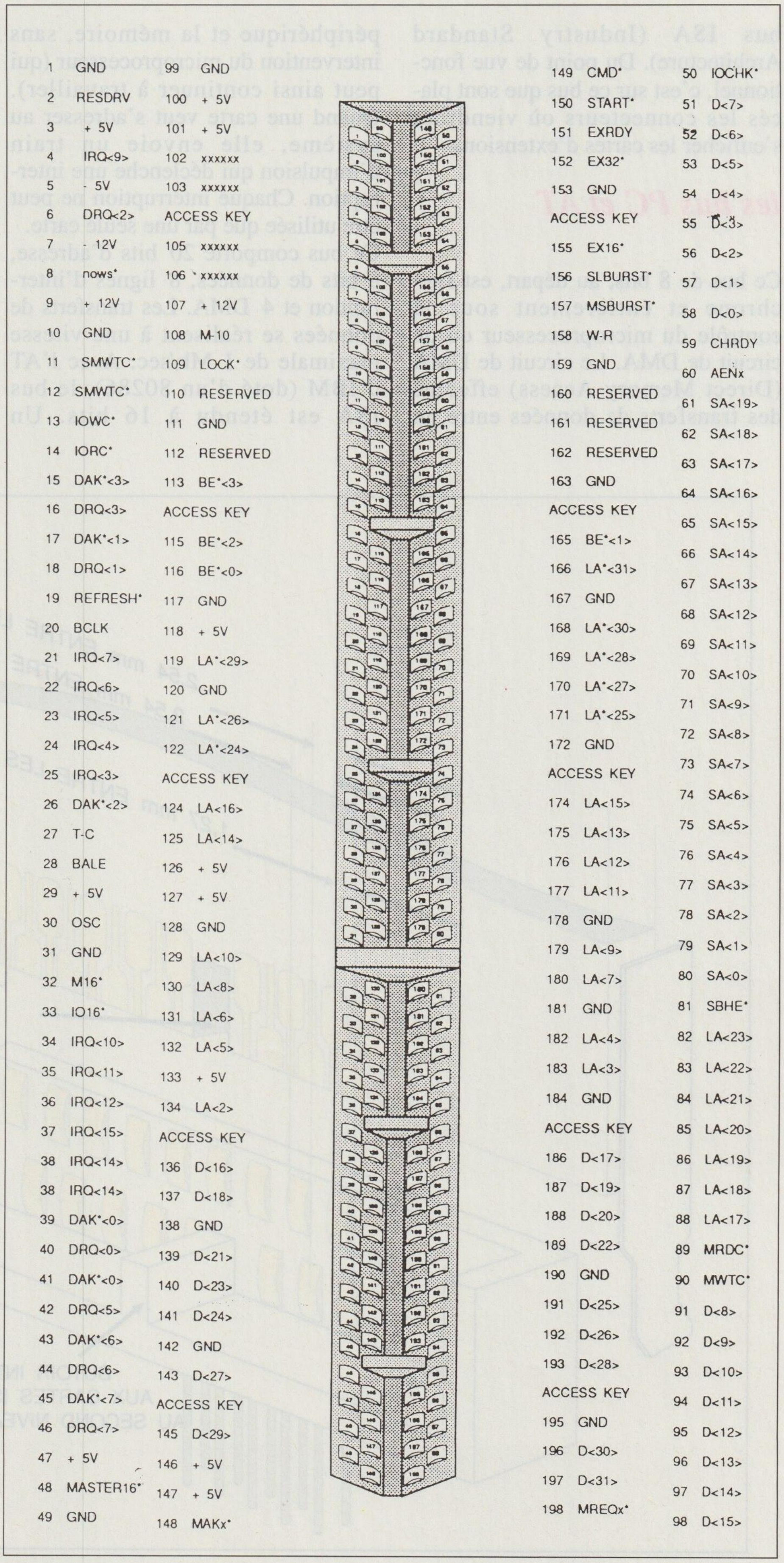
Ce bus de 8 bits, au départ, est synchrone et entièrement sous le contrôle du microprocesseur ou du circuit de DMA. Le circuit de DMA (Direct Memory Access) effectue des transferts de données entre un périphérique et la mémoire, sans intervention du microprocesseur (qui peut ainsi continuer à travailler). Quand une carte veut s'adresser au système, elle envoie un train d'impulsion qui déclenche une interruption. Chaque interruption ne peut être utilisée que par une seule carte. Ce bus comporte 20 bits d'adresse, 8 bits de données, 8 lignes d'interruption et 4 DMA. Les transferts de données se réalisent à une vitesse maximale de 1 Mb/sec. Avec l'AT d'IBM (doté d'un 80286), le bus ISA est étendu à 16 bits. Un deuxième connecteur est ajouté juste à la suite du premier pour pouvoir accéder aux bits supplémentaires. Ce bus étant plus rapide, une gestion de temps d'attente est implémentée pour conserver la compatibilité avec les cartes 8 bits. Une broche, jusque-là inutilisée sur le connecteur 8 bits, est réservée pour des cartes 8 bits acceptant de travailler plus rapidement (sans temps d'attente, ou en anglais *« 0 wait state »*). Un mécanisme est implémenté sur les cartes 16 bits pour reconnaître plus rapidement qu'elles sont adressées et ainsi



envoyer un signal au bus pour lui indiquer de procéder par cycles d'échanges en 16 bits. Ce bus AT comporte 24 bits d'adresse, 16 bits de données, 16 lignes d'interruption et 8 DMA. La vitesse maximale de transfert atteint 4 Mb/sec. Le bus EISA est une extension du bus AT à 32 bits. Il comporte 32 bits d'adresse et 32 bits de données. Cependant, les changements ne se limitent pas à une simple extension du bus de données.

vient l'EISA et le bus master

La première innovation, sûrement la plus importante, est le « bus ma ster ». Cette fonctionnalité, déjà introduite par IBM avec le bus MCA, permet à une carte de prendre le contrôle du bus (contrairement au bus ISA, où le contrôle du bus était réservé au CPU ou au circuit de DMA. Les cartes capables de prendre le contrôle du bus (souvent qualifiées de cartes « intelligentes ») contiennent un microprocesseur qui prend en charge la gestion des échanges. Ceci soulage le processeur central et permet d'obtenir, enfin, des cartes actives. Bien entendu, si plusieurs cartes « intelligentes » sont installées sur le bus, un mécanisme d'arbitrage est nécessaire. Celui-ci doit décider, à un instant donné, quelle carte peut prendre le contrôle du bus. Ce mécanisme est conçu de telle manière qu'il n'utilise pas le bus et n'empêche en rien les transferts. Un système de priorité hiérarchique permet de savoir à qui allouer le bus (voir figure). Si le bus est utilisé lors d'une demande de prise de contrôle, le système arbitre masque le signal de prise de contrôle du demandeur et lui envoie un signal lui indiquant que le bus est occupé. Dès que le bus est libéré, il renvoie au demandeur un signal lui indiquant que le bus est libre. De plus, pour éviter qu'un demandeur monopolise le bus, un temps maximum est alloué à chaque prise de contrôle. Une fois le temps écoulé, le contrôle du bus est donné au demandeur suivant en



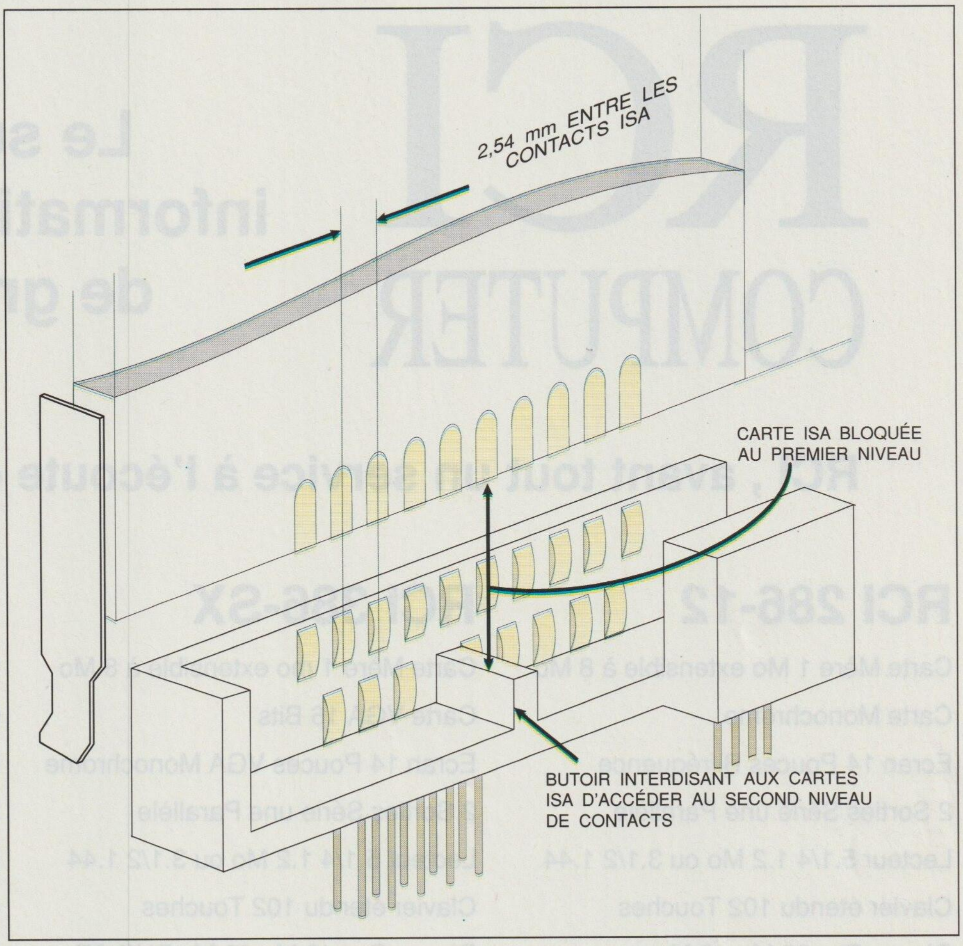
respectant la hiérarchie des priorités. Pour le bus ISA, chaque carte utilisait une ligne d'interruption.

Avec le bus EISA, les lignes d'interruption peuvent être partagées. La demande d'interruption n'est plus réalisée par un train d'impulsion mais par un niveau logique. La détection de la carte ayant lancé l'interruption doit s'effectuer au niveau logiciel. Cependant, seules les cartes EISA peuvent partager une même interruption. L'intérêt du partage est de pouvoir doter un PC de plus de cartes que de lignes d'interruption.

le partage des lignes d'interruption et le DMA

Les fonctions de DMA ont été étendues pour accélérer les transferts. Les échanges peuvent porter sur 8, 16 ou 32 bits. Quatre modes de transfert sont disponibles. Le premier est entièrement compatible avec ISA. Le deuxième, légèrement plus rapide, reste compatible avec la plupart des cartes ISA. Le troisième, nettement plus rapide, n'est accepté que par quelques cartes ISA. Ces deux derniers modes permettent des transferts en 32 bits avec les cartes EISA. Le dernier mode est le transfert par rafale (« burst »), réservé aux seules cartes EISA. On obtient, dans ce mode, des taux de transfert de 8,2 Mb/s en 8 bits, 16,5 Mb/s en 16 bits et 33 Mb/s en 32 bits Avec trois formats de données, on peut se demander comment les différentes cartes parviennent à communiquer ?

La gestion est entièrement réalisée par les circuits du bus. Ces circuits réalisent la séparation des octets et l'envoi — dans le format approprié — à la carte réceptrice. Ainsi, lorsqu'une carte de 16 bits reçoit des données d'une carte de 32 bits, cette dernière se contente d'envoyer ses données au format 32 bits. Les circuits du bus effectuent la transformation et émettent les données au format 16 bits.



configuration par logiciel

Un autre avantage des cartes EISA est d'être entièrement configurable par logiciel (plus de problème de « switch »). Ceci permet de reconfigurer un système sans l'ouvrir. Les différentes fonctions de la carte sont répertoriées dans une aide en ligne et peuvent être plus complexes qu'avec une configuration par switch (surtout pour les cartes « intelligentes»). Le logiciel détecte les conflits intervenus sur le bus et permet de les résoudre. De plus, grâce à un fichier de configuration, l'utilitaire peut indiquer comment positionner les switchs sur une ancienne carte ISA pour obtenir une configuration compatible avec le système. Le connecteur EISA comporte deux niveaux superposés de contacts. Le premier (le plus haut) est identique au connecteur ISA de 16 bits. Le second niveau (celui du bas) comporte tous les contacts nécessaires aux nouvelles fonctionnalités d'EISA. Des taquets empêchent les cartes ISA d'être enfoncées trop bas. Des encoches dans les cartes EISA correspondant aux taquets permettent d'enficher les cartes EISA jusqu'au fond du connecteur. Pour l'utilisateur, le bus EISA présente des avantages certains. Grâce à la compatibilité avec le bus ISA, le changement de bus ne limite en rien son choix de cartes d'extension. Il n'est pas non plus dépendant d'une seule marque et il peut faire jouer la concurrence. Pour les concepteurs de périphériques, ce bus permet de développer de nouvelles cartes de manière sereine, sans être obligé de concevoir toute une gamme pour la nouvelle machine. D'ailleurs, le but n'est pas de transformer toutes les cartes actuelles au format EISA, mais plutôt de ne développer que des cartes nécessitant des transferts rapides ou manipulant de grosses quantités d'informations, tels que les contrôleurs de disques ou les cartes réseaux.

Vincent Pouilley

RCI COMPUTER

Le sur mesure informatique à des prix de grande série

RCI , avant tout un service à l'écoute de vos besoins

RCI 286-12	RCI 386-SX	RCI-386-25
Carte Mère 1 Mo extensible à 8 Mo	Carte Mère 1 mo extensible à 8 Mo	Carte Mère 2 Mo extensible à 8 Mo
Carte Monochrome	Carte VGA 16 Bits	Carte VGA 16 Bits
Ecran 14 Pouces Bifréquence	Ecran 14 Pouces VGA Monochrome	Ecran VGA Multisync
2 Sorties Série une Parallèle	2 Sorties Série une Parallèle	2 Sorties Série une Parallèle
Lecteur 5.1/4 1.2 Mo ou 3.1/2 1.44	Lecteur 5.1/4 1.2 Mo ou 3.1/2 1.44	Lecteur 5.1/4 1.2 Mo et 3.1/2 1.44
Clavier étendu 102 Touches	Clavier étendu 102 Touches	Clavier étendu 102 Touches
Disque Dur 20 Mo 40 Ms	Disque Dur 40 Mo 28 Ms BUS AT	Disque Dur 40 Mo 28 Ms BUS AT
Carte Controleur Interleave 1/1	Carte Controleur Interleave 1/1	Carte Controleur interleave 1/1
Souris Compatible	Souris Compatible	Souris Compatible
MS DOS 4.01	MS DOS 4.01	MS DOS 4.01
9200 F TTC	13850 F TTC	24950 F TTC

RCI 486-25

A partir de 36200 F

Matériel garanti 2 ans Pièces et Main D'oeuvre

Afin de recevoir une documentation

Téléphonez ou écrivez à

RCI COMPUTER

67 Rue de la Jonquière

75017 PARIS

Tel : 42.26.70.04 Fax: 42.26.74.84

RCI
Pensez-y

RCI
Pensez-y

Cerclez info pub 26

Initiation à la bureautique

AU CŒUR DU PPP

Fin de la présentation du Programmeur Personnel de Paradox : les fonctions avancées, comment modifier une application.

(Le PPP. Seconde partie)

Le premier volet de cette étude du PPP a permis de mieux cerner la nature et le champ d'application de ce module de Paradox 3.0. Quelques mots suffisent à le définir de manière acceptable : il s'agit d'un générateur permettant de développer des applications spécifiques sans nécessiter l'écriture de la moindre ligne de code. Cette vision idyllique — que tous les générateurs d'applications revendiquent — ne correspond pas tout à fait à la réalité. Une application développée à l'intérieur de ce cadre strict, revêt un caractère limité. Elle comporte tout au plus des menus personnalisés, associés à des fonctions élémentaires : appel d'un format d'impression, d'une fiche de saisie, etc. Le recours à des scripts permet certes d'accroître sensiblement le champ d'application du PPP, et d'automatiser un certain nombre de tâches. En contre-partie, cela implique une connaissance du langage de programmation de Paradox 3.0 (PAL), et donc l'écriture de lignes de code.

user sans abuser

Pour un utilisateur averti de Paradox, il ne saurait être question d'abuser du PPP. Non pas pour des raisons de morale, mais plutôt dans un souci d'efficacité et de performance. Dans cette optique, l'attrait majeur du PPP réside sans conteste dans son aptitude à générer des menus personnalisés, parfaitement structurés et exploitables par un développeur. La structure arborescente de l'application, les niveaux successifs de menus et de sous-menus, et les actions associées (déclarées ou placées « en attente »), représentent une structure idéale, parfait point de départ pour concevoir une application complexe. A chacun, ensuite, d'habiller la structure et de finaliser l'ouvrage, en intégrant les scripts et les procédures prévus à cet effet.

Au nom de la productivité, mieux vaut concevoir la majorité des éléments constitutifs de l'application (tables, fiches et requêtes multitables, formats d'impressions, et scripts en particulier) en dehors du PPP. De cette façon, la session du PPP peut être consacrée aux seules opérations d'assemblage, sans digression ni égarement.

Mieux, la fonction « Résumer », inscrite au menu principal du PPP, permet d'obtenir une documentation exhaustive de l'application :

— la structure des menus, avec la liste des fonctions et des objets associés, les chemins d'accès et le nom des tables associées,

— la liste complète des fichiers utilisés par l'application,

— une vue de l'arborescence des menus,

— les trois états précédents simultanément.

Cette option prend tout son sens dans le cas d'applications volumineuses, en cours d'élaboration ou nécessitant des aménagements. L'utilisateur retrouve sans mal la trace d'un script défaillant ou d'un menu incomplet.

S'il est préférable de créer l'ensemble des éléments de l'application en dehors du PPP, celui-ci offre toutefois l'opportunité d'éditer un script, de définir un format d'impression, et même de générer une nouvelle table, si le besoin s'en fait sentir. Une « roue de secours » bienvenue en quelque sorte.Les options de requête et d'impression, quant à elles, font plus que dépanner le programmeur : elles lui facilitent la vie. La commande « Vue » permet, par exemple, de spécifier les champs à afficher, ou encore d'afficher une partie seulement des enregistrements de la table, sélectionnés à partir d'une fiche de requête QBE. Ces différentes opérations se passent de fiches prédéfinies et de scripts de requête. Précisons toutefois qu'elles ne peuvent pas être activées en mode édition, où l'ensemble des enregistrements apparaît, et où les champs retenus dépendent de la fiche de saisie utilisée. Le mode édition, qui permet de modifier les données exis-

tantes, échappe à la restriction : le développeur peut occulter une partie des enregistrements. Toutefois, pour des raisons d'intégrité des données et d'organisation de Paradox, les champs clés ne pourront subir de modification.

Les menus générés par le Programmeur Personnel appellent parfois des fonctions d'impression. Le PPP se rapproche alors de sa définition première : il remplace alors avantageusement l'écriture de code. Il suffit au programmeur de spécifier un certain nombre de paramètres élémentaires (tables concernées, format d'impression souhaité, périphérique cible), de remplir une feuille de requête QBE s'il désire obtenir l'impression d'une partie seulement des enregistrements, et le PPP se charge du reste. Il commande le tri des enregistrement, copie le format d'impression (ou rapport) de la table source vers la table réponse (issue de la requête), commande l'impression vers le périphérique adéquat (imprimante, écran, disque). Le PPP se substitue entièrement à un éventuel script qui comprendrait la requête, la commande DO_IT! pour la mettre en oeuvre, et une séquence destinée à affecter un rapport prédéfini à la table réponse (COPYREPORT...).

un PPP nourri au PAL

L'oubli, l'étourderie, et les caprices des clients étant de fidèles compagnons des programmeurs, les possibilités de corrections et de reprises des applications conçues à partir du PPP constituent un aspect non négligeable du problème. Une fonction comme « résumer », décrite un peu plus haut, offre ainsi une vue générale et précise de l'organisation de l'application. Le maillon défaillant est rapidement repéré. Puisque le développement se déroule en plusieurs étapes, il est naturel de pouvoir laisser une commande d'un menu ou d'un sous-menu en attente d'affectation. Ces remaniements peuvent être réalisés avec bonheur en utilisant le PPP. Paradox 3.0 prend en compte les changements apportés — primordial pour préserver l'efficacité de la commande « résumer » — et ne récrit que les seules parties de l'application concernées.

L'autre méthode de correction des applications consiste à agir au niveau du code. Le PPP traduit en effet les applications en langage PAL, spécifique à Paradox. La clarté et la rigueur de ce code — lignes indentées et commentaires en particulier — permettent à tout habitué du PAL de s'y retrouver aisément.

Ces « bidouillages » présentent un avantage indéniable : ils constituent la solution idéale pour contourner les limites du PPP. Ils s'imposent donc dans un certain nombre de situations. La première étape suivant le lancement d'une application (créée via le PPP) consiste à afficher un message invitant l'utilisateur à saisir un mot de passe. Il limite l'accès à certaines opérations sur les tables de la base, pour peu que celles-ci soient effectivement protégées par un système de mot de passe. Si l'on se fie au PPP, il est impossible d'éviter ou de compléter cette phase préliminaire. Pour ce faire, il convient de récrire les quelques lignes de code concernées, contenues dans le fichier [Nom de l'application].SC — appelé script pilote de l'application. L'unique limite aux changements envisageables est l'imagination (et les impératifs) des développeurs, comme le montre modestement la figure 1.
D'autres raisons peuvent vous amener à modifier le code, en voici quelques-unes, accompagnées du nom du fichier abritant le fragment de programme à retoucher — le symbole # figure un nombre et * le nom de l'application.

— modifier l'écran de présentation (ajout de couleurs par exemple) : *G##.SC,
— modifier les écrans d'aide : *H##.SC,
— traduire les messages d'erreurs en cas d'utilisation d'une version américaine de Paradox : *.SC,
— modifier rapidement les lignes de commentaires attachées aux sélections des menus, ou ajouter des choix supplémentaires à un menu : *##.SC,
— substituer une commande mono-utilisateur (EDIT par exemple) par

Un Kit de saisie nébuleux

Le Kit de saisie des données de Paradox 3.0 est sans conteste l'outil le plus performant offert aux développeurs.
Il se compose d'un certain nombre de procédures et d'utilitaires. Deux d'entre eux se révèlent particulièrement intéressants :
— DoWait, l'élément principal du Kit, permet de contrôler intégralement la saisie des données. Le programmeur peut affecter des procédure en entrée ou en sortie de champ, selon la nature des données saisies (si telle valeur est entrée alors appeler telle procédure, ou se rendre vers tel champ, etc.),
— Popup 2 permet de définir des menus verticaux, se déroulant à l'intérieur de la fenêtre paramétrée, à partir desquels l'utilisateur peut sélectionner une valeur.

Les possibilités étendues du Kit de saisie n'ont d'égal, malheureusement, que sa complexité de mise en oeuvre. Pour espérer en tirer le moindre parti, il est nécessaire de maîtriser PARFAITEMENT les notions de procédures, de bibliothèques de procédures, et de variables locales et globales. Sans oublier, une connaissance exhaustive du langage PAL, et des rouages internes de Paradox. Vaste programme convenez-en !

Un prochain chapitre d'« Initiation à la bureautique » tentera de démêler l'écheveau du Kit de saisie des données.

J.-F. B.

```
; ask for the password to the application
; this password determines the access to the tables
; in the application allowed for
; the current user of the application.

@ 0, 0
Style Attribute SysColor(0)
?? fill(" ",160)
@ 1, 0
?? "Entre le mot de passe; [Esc] pour abandonner"
@ 0, 0
?? "Mot de passe: "
Cursor Normal
zzzcolor = int(SysColor(0) / 16)
Style Attribute ((zzzcolor * 16) + zzzcolor)
Accept "a50" To pword
Style
EscEnter = not retval
Cursor Off
if (EscEnter) then
 Message "Quitter l'application"
  Sleep 2000
  Clear
  return
endif
if (pword <> "") then
 Password pword
endif
..........................................................
Clearall
if (pword <> "") then
 UnPassword pword
endif
```

Conçue à partir de Paradox 3.0 en version américaine, l'application offre des commentaires en anglais. La demande de mot de passe est tout à fait formelle et ne conditionne pas le lancement de l'application.

```
; Demander un mot de passe à l'utilisateur.
; Ce mot de passe déterminera si l'application
; doit être chargée ou non.
; Mémorisé, il servira à restreindre
; l'accès à certaines tables de la base.
@ 0, 0
Style Attribute SysColor(0)
?? fill(" ",160)
@ 1, 0
?? "Entrer le mot de passe [Esc] pour abandonner"
@ 0, 0
?? "Mot de passe: "
Cursor Normal
zzzcolor = int(SysColor(0) / 16)
Style Attribute ((zzzcolor * 16) + zzzcolor)
Accept "a10" To pword
Style
EscEnter = not retval
Cursor Off
if (EscEnter) then
  Message "Quitter l'application"
  Sleep 2000
  Clear
  return
endif
if (pword <> "") then
  if pword <> "actionpc" then
    Then exit
    else Password pword
endif
..........................................................
Clearall
if (pword <> "") then UnPassword pword
endif
```

Outre le commentaire en français, le bout de script a été modifié de manière à interdire l'accès à l'application si le mot de passe adéquat (actionpc) n'est pas saisi.

figure 1 : exemple de modification d'une application en agissant au niveau du code PAL.

une commande multiutilisateur (COEDIT).

Attention toutefois ! Si certains de ces fichiers peuvent être modifiés naturellement comme tout script de Paradox (*.SC ou *G##.SC), d'autres sont constitués de procédures (*##.SC). Il faut alors non seulement sauvegarder le script modifié, mais en plus récrire la bibliothèque de procédure.

Deux solutions se présentent

« jouer » le script en question qui contient la commande WRITELIB, ou déclencher le script *LIB.SC qui se charge de recomposer l'ensemble des bibliothèques de procédures gérées par l'application.

Au moment d'entreprendre de telles manipulations, songez à respecter quelques règles de base. En premier lieu, gardez à l'esprit qu'un changement (l'intitulé d'une sélection d'un menu par exemple) peut entraîner une faillite de l'application s'il n'est pas répercuté chaque fois que l'élément considéré intervient. Veillez également à ce que les variables n'interfèrent pas entre elles.

Enfin, ne perdez pas de vue une réalité implacable : les corrections « manuelles » apportées à une application - en dehors du PPP donc - ne survivront pas à une session de modification par l'entremise du PPP. Celui-ci récrit le code de l'application en éliminant les commandes parasites greffées par le programmeur

Jean-François Balaine

LA BOUTIQUE DE PRESSIMAGE

210 rue du faubourg Saint Martin 75010 PARIS

LOGICIELS PC ET COMPATIBLES (5"1/4 ou 3"1/2)

Ces logiciels sont des Freeware, certains en anglais. Par conséquent la Boutique de Pressimage n'assure pas de support technique sur ces produits. Seules les disquettes défectueuses seront échangées. Attention certaines polices de caractères ne sont pas accentuées !!

❑ **LASERJET FONTS 1** ***(100 frs)***
Les deux disquettes contiennent des polices de caractères pour HP Laserjet,
- *Century Schoolbook* 8 pt normal, 10 et 12 pt, normal, gras, et italique, 18 et 24 pt normal (portrait).
- *Garamont* 12 pt normal, gras, italique, 18, 24 et 30 normal (portrait).
- *Optima* 10 et 12 pt normal, gras et italique, 18 et 24 pt normal (portrait).
- *Times-Roman* 12 pt normal, gras, et italique, 12 pt gras italique ainsi que 12 pt gras élargi (portrait).
Nécessite une imprimante HP Laserjet Plus / II.
Egalement sur cette disquette un utilitaire de téléchargement.

❑ **LASERJET FONTS 2** ***(100 Frs)***
Ces deux disquettes contiennent des polices de caractères pour HP Laserjet :
- *Pica* 10 pt normal, gras et italique (portrait et paysage).
- *Old English* et *Script* 18 pt (portrait).
- *Computer Symbols* et *Zapf dingbats* 10 pt (portrait).
- *Prestige* 8 pt normal.
- *Garamont* 10 et 12 pt normal, gras et italique (portrait et paysage).
- *Garamont* 18 pt normal et gras.
Nécessite une imprimante HP Laserjet Plus / II.
Un utilitaire de téléchargement est fourni.

❑ **LASERJET UTILITIES 1** ***(50 Frs)***
Cette disquette contient un menu d'installation des utilitaires pour les imprimantes HP Laserjet Plus et II. Ce menu peut être résidant en mémoire. Il contrôle les fonctions de l'imprimante comme la sélection d'une fonte ou le choix de la marge. Inclus également, un utilitaire qui permet de transformer les fontes pleines en fontes contour ou 3D. Un troisième utilitaire permet d'imprimer des enveloppes. Il n'utilise pas la cartouche de fontes, mais a besoin de la police paysage interne de 10pt. Nécessite une imprimante HP Laserjet Plus / II.

❑ **LASERJET UTILITIES 2** ***(50 Frs)***
Cette disquette contient *PAMPHLET* qui permet d'utiliser le mode paysage et de placer deux pages côte à côte, *HPL2VEN* convertit les polices HP au format VENTURA, *GEMCAP* qui capture les images écran pour VENTURA.
LJPRINT imprime deux pages entières en réduction sur un format 11", plus quelques autres utilitaires.

❑ **DESKJET FONTS** ***(50 Frs)***
Cette disquette contient quelques fontes pour l'imprimante HP Deskjet.
- *Helvetica* 10 pt médium gras, italique et romain.
- *Times-Romain* 10 pt.
- *Script* 10 pt.
- *Symbols* 24 pt, Copyrights, puces, chiffres encerclés et beaucoup plus. Inclus également, un utilitaire qui imprime une page exemple de fontes téléchargeables pour la Laserjet ou la Deskjet, et un programme qui examine les fontes de ces imprimantes et imprime une page d'informations.

❑ **PC-WRITE V3.02** ***(150 Frs)***
C'est un traitement de texte complet, en trois disquettes, contenant les options suivantes :
multicolonnage avec visualisation à l'écran, ligne d'état en haut de l'écran, aide en ligne sur plusieurs écrans, menus plein écran pour les polices et les effets, dictionnaire orthographique anglais rapide, coupe, copie, collage et effacement de blocs de texte. Il supporte parfaitement le clavier étendu, sauvegarde automatique paramétrable, et beaucoup d'autres possibilités qui font de ce logiciel l'un des meilleurs.

❑ **EZ FORMS EXECUTIVE** ***(50 Frs)***
Un générateur de formulaires contenant plus de 125 écrans de menus et d'instructions complètes, un mode tutorial et un système d'aide interactif. On y trouve les options suivantes : feuille de calcul mathématique, fusion de fichiers, tabulation automatique, opérations de bloc, gestion des recouvrements, support étendu d'impression vue à l'écran, et beaucoup plus. La version Shareware imprime plus lentement. Ce logiciel nécessite 512 Ko et un imprimante matricielle.

❑ **FORMGEN V 3.11** ***(50 Frs)***
Générateur de formulaires géré par des menus et un système d'aide en ligne. Ce logiciel possède toutes les options nécessaires à la création de formulaires personnalisés. Il nécessite 353 Ko de mémoire vive ainsi que imprimante matricielle.

❑ **MR LABEL** ***(50 Frs)***
Un programme dédié à la conception d'étiquettes, et qui vous permet d'utiliser votre imprimante comme une machine à écrire, ou de retrouver et d'imprimer les informations d'autres programmes. Quelques options : plus de 9 étiquettes de front, reconnaissance de toutes les fontes supportées par votre imprimante, centrage automatique, multicopie, possibilité de dessiner vos propres étiquettes, changement de fonte à chaque ligne, etc.
Nécessite 128 Ko de mémoire vive, ainsi qu'une imprimante matricielle.

❑ **LABEL MASTER** ***(50 Frs)***
Voici un logiciel de publipostage, puissant, mais néanmoins facile à utiliser. Vous pouvez créer votre fichier de noms et d'adresses, l'éditer, y rechercher un enregistrement particulier, vous déplacer à travers ce fichier, et imprimer des étiquettes en n'importe quel format. D'autres options : champs mémo pour commentaires, Pays et états/champs province pour les pays étrangers, système d'aide par menu pop-up, calendrier, calculatrice, filtrage des enregistrements, saisie automatique au format ville/état/pays et une documentation complète.
Nécessite 384 Ko de mémoire vive, ainsi que deux lecteurs.

❑ **IMAGEPRINT V1.5** ***(50 Frs)***
Ce logiciel permet d'imprimer en qualité courrier avec une imprimante 9 aiguilles compatible EPSON, mode graphique IBM et plus.

❑ **SIDEWRITER V3.01** ***(50 Frs)***
Ce logiciel permet l'impression à l'italienne sur une grande longueur de papier. Les colonnes d'une feuille de calcul d'un tableur peuvent être imprimées sur une page continue. C'est parfait pour Lotus 123 ou n'importe quel fichier ASCII.
Nécessite 128 Ko de mémoire vive ainsi qu'une imprimante compatible EPSON / IBM.

❑ **FINGERPAINT V2.00** ***(50 Frs)***
C'est un logiciel de dessin similaire à MacPaint. Il utilise le clavier ou la souris, et ses drivers sont résidants en mémoire. Six fontes de caractères sont disponibles en 81 tailles différentes. Utilisé pour la PAO, les slide-show et la fabrication de dessins animés. La version enregistrée auprès de l'éditeur supporte les imprimantes matricielles, et la version shareware uniquement l'imprimante HP Laserjet. Nécessite 320 Ko de mémoire vive, un moniteur CGA, VGA ou Monochrome.

❑ **PRINT SPOOLERS** ***(50 Frs)***
Cette disquette contient 4 spoolers d'impression différents. Disk spool ? Spool, vous permet d'imprimer directement et de trier vers un fichier disque. Spool spooler en RAM dont la taille est ajustable. Spool spooler en RAM possède des options différentes du précédent spooler.

❑ **PS-TYPE STUDIO V1.30 (50 Frs)**
Ce programme est un générateur de fontes logicielles pour Xerox Ventura Publisher installé avec une imprimante Postcript. Il permet de fabriquer une infinité de nouvelles fontes ; condensées, élargies, ou entourées avec des épaisseurs variables de ligne et de contour. Il permet également d'ombrer les caractères avec n'importe quel pourcentage de gris. Remplir les caractères en dégradé, effectuer une rotation indifféremment sur la ligne de base ou sur l'origine du caractère, incliner, étroitiser, entourer ces derniers avec jusquà deux ombres, appliquer jusqu'à six niveaux d'ombres portées, est un véritable jeux d'enfant... Il génère automatiquement les fontes écran ainsi que les gras, les italiques, et les gras italiques. Ce logiciel est compatible avec Ventura 2.0 et ses extensions.

❑ **VENTURA DESKTOP UTILITIES V2** ***(50 F)***
Cette disquette contient six utilitaires pour travailler avec Xerox Ventura Publisher. *TAB* convertit les espaces de tabulation. Il est parfait pour convertir les fichiers 123.prn et les autres tables pour Ventura. *NEWLINE* convertit des fichiers ASCII et restitue leurs fins de ligne et leurs fins de paragraphe réelles. Vos lignes ne sont colleront plus les unes aux autres, lors de l'importation sous Ventura, et vos paragraphes resteront ensembles. *STRIPSPACE* extirpe les espaces exédentaires d'un fichier ASCII. *VP-DEL* supprime en un éclair des chapitres et/ou des fichiers qui y sont associés. *PS-PRN* imprime in fichier ASCII sur une laser Postcript sans passer par l'émulation Diablo. *BACKIT* permet de fragmenter un gros fichier sur plusieurs disquettes en vue d'un transport, et ce qui est plus important de les restaurer sur un disque dur DOS en utilisant la commande DOS : COPY. Parfait pour transférer des fichiers images gigantesques.

❑ **1ST PUBLISHER CLIP ART/2** ***(50 Frs)***
Contient 86 planches de dessins sur les ordinateurs, la médecine et la musique. Nécessite 1ST PUBLISHER.

❑ **TOP FORTY N° 16** ***(50 frs)***
Contient des masques de lettres et de formulaires courants... A décompacter sur 2 disquettes double-face et une simple face (décompacteur fourni).

❑ **TOP FORTY N° 19 (50 Frs)**
Des polices de caractères pour les imprimantes HP Laserjet. Elle contient des fichiers archive à décompacter sur cinq disquettes double face (décompacteur fourni).

❑ **TOP FORTY N° 21 (50 Frs)**
Encore une foule d'utilitaires pour HP Laserjet à décompacter sur 3 disquettes simple face (décompacteur fourni).

❑ **TOP FORTY N° 29 (50 Frs)**
Cette disquette contient FLU-SHOT+, un antivirus pour les disques durs ainsi que beaucoup d'autres antivirus pour les disquettes.

La Boutique de Pressimage c'est aussi des logiciels ATARI AMIGA MACHINTOSH
pour tout renseignement contactez nous au 42.49.21.97.

LA BOUTIQUE DE PRESSIMAGE

210 rue du faubourg Saint Martin 75010 PARIS

DECOUVREZ UNE FOULE D'UTILITAIRES POUR VOTRE PC ALLANT DU TRAITEMENT DE TEXTE AU LOGICIEL DE DESSIN EN PASSANT PAR L'ANTI-VIRUS... ET COMME TOUJOURS LA 5ème DISQUETTE EST GRATUITE

SPECIAL PC

consultez notre descriptif (avant les pages abonnement et service lecteurs)

logiciels sur disquettes 5"1/4 à 50frs et sur disquettes 3"1/2 à 75 frs

BON DE COMMANDE

J'ai connaissance du fait que certains produits ont une notice ou un écran en langue anglaise et je désire les acquérir. Veuillez expédier ma commande à l'adresse suivante:

NOM :...

PRENOM :...

ADRESSE :...

...

CODE POSTAL :...

VILLE :...

DATE :/......./1990

SIGNATURE: (celle des parents pour mineurs)

Ci-joint mon règlement à l'ordre de PRESSIMAGE

- ☐ Chèque
- ☐ C.C.P

Etranger, nous consulter.

DISQUETTES	QTÉ	PRIX

disquettes 5"1/4 ☐
disquettes 3"1/2 ☐

FRAIS DE PORT	15 frs
TOTAL	

BON DE COMMANDE A RETOURNER A:
LIBRAIRIE PRESSIMAGE / ACTION PC
210 rue du Faubourg Saint Martin
75010 PARIS

LES LIAISONS DANGEREUSES

La communication entre matériels de différentes marques a toujours été un des problèmes majeurs de l'informatique. Les solutions actuelles sont devenues fiables et efficaces.

Lorsque l'informatique est née, les ordinateurs n'avaient que fort peu besoin de communiquer et, lorsque tel était le cas, c'était généralement entre appareils d'une même marque tant les fabricants étaient peu nombreux. Les liaisons étaient alors très souvent de type spécialisé et incompatibles d'une marque à une autre.

Peu à peu, le besoin de communication s'est fait plus pressant et un certain nombre de normes ont vu le jour. Bien qu'elles ne soient pas parfaites, elles permettent aujourd'hui de relier et de faire dialoguer des appareils très divers.

Cette série d'articles se propose de vous présenter les solutions les plus usitées dans l'environnement PC / PS et de fournir un certain nombre de méthodes permettant de résoudre la majorité des problèmes de compatibilité que l'on rencontre.

La théorie et la partie électronique ont été réduites à la portion congrue afin que chacun puisse aborder ces articles sans peine.

les divers types de liaisons disponibles

Rappelons que tous les équipements informatiques ne manipulent que des informations binaires, c'est-à-dire représentées par des 0 et des 1. Cette manipulation se fait au moyen de circuits logiques en technologie TTL, CMOS ou encore ECL selon la classe et la rapidité de la machine concernée. Les circuits TTL - ou aux normes TTL - étant encore les plus répandus (particulièrement dans les machines de type PC ou PS), ce sont eux qui illustreront nos propos. Dans de tels circuits, l'information binaire est codée par des tensions. Un 0 logique est représenté par une tension comprise entre 0 et 0,8 volt, tandis que toute tension comprise entre 2,4 volts et 5 volts est un 1. Les tensions comprises entre 0,8 volt et 2,4 volts ne doivent pas, en principe, exister puisqu'elles ne correspondent à aucune valeur logique. Pour atteindre cet objectif, la transition entre deux états logiques est extrêmement rapide (une vingtaine de nanosecondes).

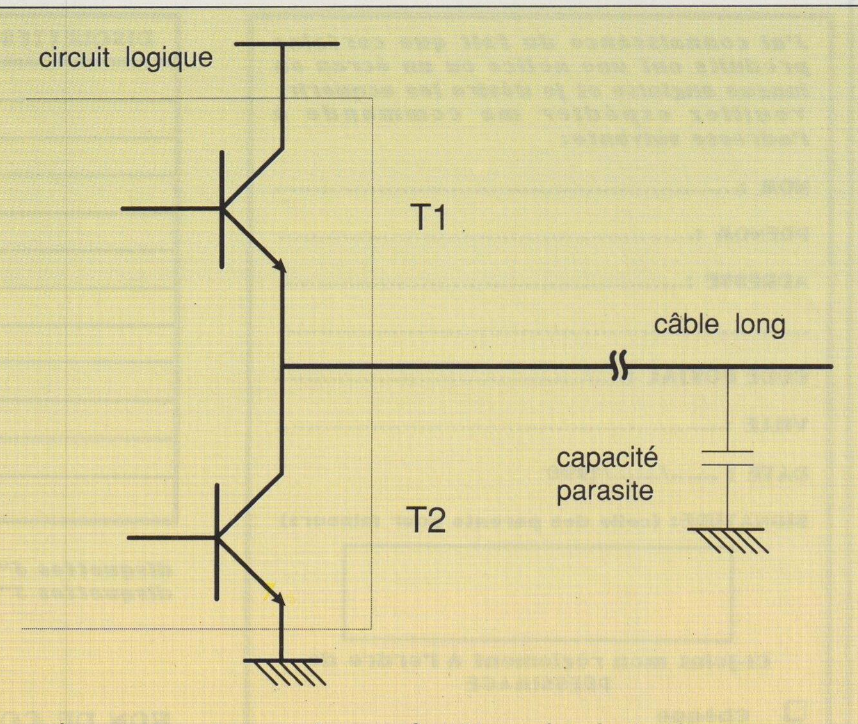


Figure 1 : étage de sortie d'un circuit logique TTL qui attaque un câble long

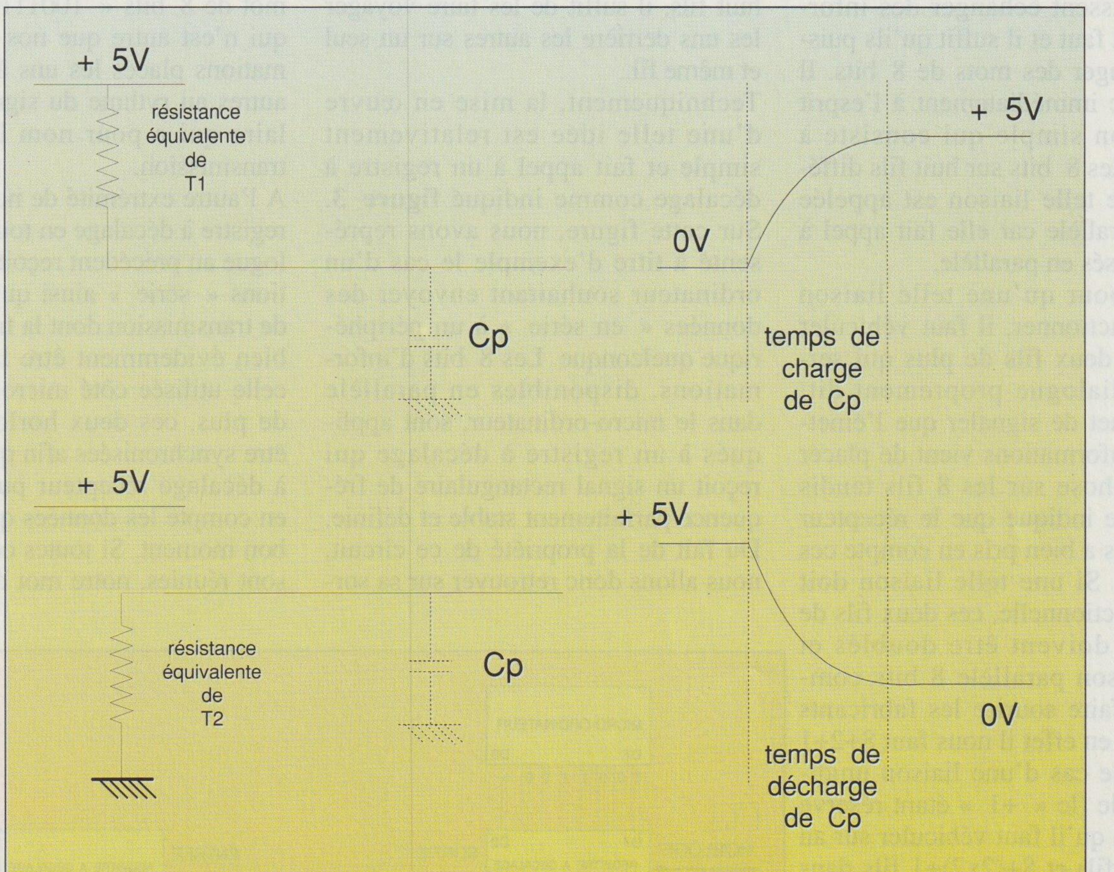


Figure 2 : dégradation des signaux en sortie du montage de la figure 1

Tant que de tels signaux ne doivent pas voyager sur des distances trop importantes, il n'existe aucun problème, mais dès que l'on dépasse le mètre ou un peu plus, les capacités parasites de câblage commencent à faire sentir leur influence et dégradent la forme des signaux.

Examinons la **figure 1** qui représente l'étage de sortie d'un circuit logique TTL qui attaque un câble de « grande » longueur. Ce câble est assimilable, pour les besoins de notre exposé, à un condensateur représentant sa capacité parasite. Lorsque nous voulons transmettre un 1 logique sur ce câble, le transistor T1 va conduire et T2 va être bloqué, notre schéma pourra donc être représenté comme sur la **figure 2** et le signal logique, au lieu d'avoir une transition bien raide entre le 0 et le 1 prendra une forme exponentielle due à la charge de la capacité parasite par la résistance équivalente au transitor T1.

De la même façon, pour transmettre un niveau logique 0 sur ce câble, c'est cette fois-ci T2 qui devra être conducteur et qui sera donc assimilable à une résistance, T1 restant bloqué. Là encore, notre signal logique ne sera pas satisfaisant car il aura toujours une transition de forme exponentielle due, cette fois, à la décharge de la capacité parasite dans la résistance équivalente de T2.

Ces temps de montée et de descente des signaux ne dépendent que de deux facteurs : la valeur de la capacité parasite et la valeur des résistances équivalentes des transistors. Sur le premier facteur, il est impossible d'agir au-delà de certaines limites puisque la capacité est due à la structure physique du câble. Quant au deuxième facteur, il est aussi très vite limité, car si l'on veut abaisser la résistance équivalente des transistors, il faut les faire travailler sous fort courant et, de ce fait, employer des modèles de puissance, ce qui est difficilement concevable dans ce genre d'application. Nous allons voir fort heureusement que d'autres solutions existent.

les liaisons parallèles

Pour que deux équipements informatiques puissent échanger des informations, il faut et il suffit qu'ils puissent échanger des mots de 8 bits. Il vient donc immédiatement à l'esprit la solution simple qui consiste à véhiculer les 8 bits sur huit fils différents. Une telle liaison est appelée liaison parallèle car elle fait appel à N fils utilisés en parallèle.
En fait, pour qu'une telle liaison puisse fonctionner, il faut véhiculer au moins deux fils de plus qui servent au dialogue proprement dit. L'un permet de signaler que l'émetteur des informations vient de placer quelque chose sur les 8 fils tandis que l'autre indique que le récepteur de données a bien pris en compte ces dernières. Si une telle liaison doit être bidirectionnelle, ces deux fils de dialogue doivent être doublés et notre liaison parallèle 8 bits commence à faire sourire les fabricants de câbles; en effet il nous faut 8+2+1 fils dans le cas d'une liaison unidirectionnelle (le « +1 » étant réservé à la masse qu'il faut véhiculer sur au moins un fil) et 8+(2x2)+1 fils dans le cas d'une liaison bidirectionnelle. Cela conduit à fabriquer de gros câbles, chers et encombrants, et ce type de liaison ne se rencontre donc que pour de courtes distances.
Son domaine d'emploi favori est celui de la connexion d'imprimantes à des ordinateurs ou micro-ordinateurs. On dit alors que l'on utilise une liaison parallèle de 8 bits ou encore une liaison « Centronics » du nom du célèbre fabricant d'imprimantes qui le premier utilisa ce type de liaison sur ses machines.

les liaisons séries

Pour réduire le nombre de fils utiles pour une liaison informatique, il n'existe qu'une solution : celle de la liaison série. L'idée est fort simple et repose sur le principe suivant : puisque l'on ne peut pas faire voyager les 8 bits d'informations simultanément étant donné que cela nécessite huit fils, il suffit de les faire voyager les uns derrière les autres sur un seul et même fil.
Techniquement, la mise en œuvre d'une telle idée est relativement simple et fait appel à un registre à décalage comme indiqué **figure 3**. Sur cette figure, nous avons représenté à titre d'exemple le cas d'un ordinateur souhaitant envoyer des données « en série » à un périphérique quelconque. Les 8 bits d'informations, disponibles en parallèle dans le micro-ordinateur, sont appliqués à un registre à décalage qui reçoit un signal rectangulaire de fréquence parfaitement stable et définie. Du fait de la propriété de ce circuit, nous allons donc retrouver sur sa sortie un signal ayant l'aspect indiqué sur cette même figure (nous avons pris l'exemple de la transmission du mot de 8 bits « 10011101 »), signal qui n'est autre que nos bits d'informations placés les uns à la suite des autres au rythme du signal rectangulaire qui a pour nom l'horloge de transmission.
A l'autre extrémité de notre câble, un registre à décalage en tous points analogue au précédent reçoit les informations « série » ainsi qu'une horloge de transmission dont la fréquence doit bien évidemment être la même que celle utilisée côté micro-ordinateur ; de plus, ces deux horloges doivent être synchronisées afin que le registre à décalage récepteur puisse prendre en compte les données qu'il reçoit au bon moment. Si toutes ces conditions sont réunies, notre mot de 8 bits sera

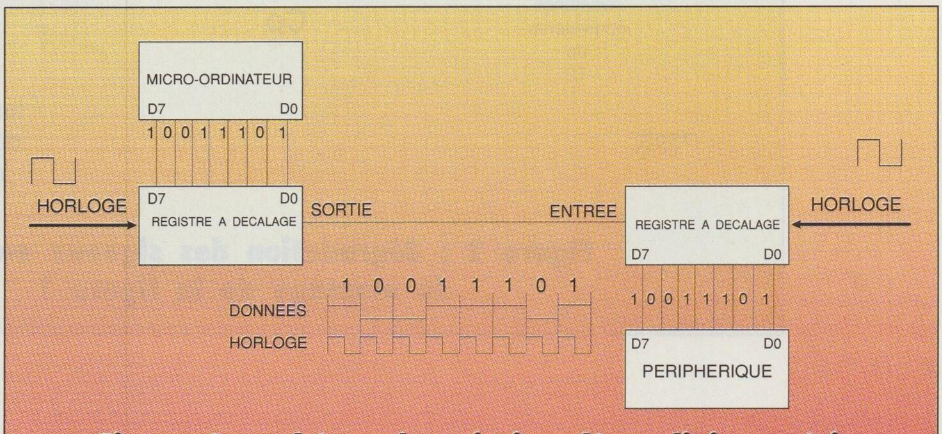


Figure 3 : schéma de principe d'une liaison série

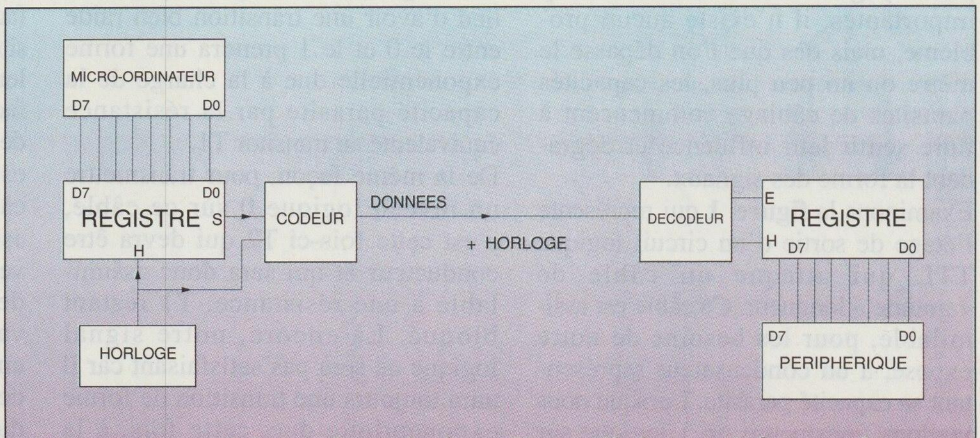


Figure 4 : schéma de principe d'une liaison série synchrone

bits				bits 7, 6, 5	000	001	010	011	100	101	110	111
				hex 1	0	1	2	3	4	5	6	7
4	3	2	1	hex 0								
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	VT	ESC	+	;	K	[	k	{
1	1	0	0	C	FF	FS	0	<	L	\	l	\|
1	1	0	1	D	CR	GS	-	=	M	]	m	}
1	1	1	0	E	SO	RS	.	>	N	^	n	~
1	1	1	1	F	SI	US	/	?	O	_	o	DEL

figure 5 : tableau officiel du code Ascii

disponible en sortie du registre à décalage récepteur sans difficulté. Par rapport à une liaison parallèle, nous avons gagné sur le nombre de fils puisque deux seulement sont utiles ici, le fil de signal que nous venons de voir et un fil de masse. Si la liaison doit être bidirectionnelle, il suffit alors de 3 fils, deux fils de données (un pour chaque sens) et un fil de masse. Le gain est appréciable, surtout sur de grandes longueurs.

En contrepartie, nous avons accru la complexité des deux montages connectés puisqu'il faut leur adjoindre un circuit dit d'interface série.

Avant d'en finir avec ces explications, il nous faut faire remarquer que l'absolu synchronisme des horloges à l'émission et à la réception impose très souvent d'utiliser la même horloge pour le registre émetteur et pour le registre récepteur ce qui conduit, soit à la véhiculer sur une fil indépendant, soit, comme représenté **figure 4**, à utiliser un codeur spécial capable de mélanger données et horloge à l'émission et à séparer données et horloge à la réception. Ce synchronisme obligatoire a fait baptiser ces liaisons : liaisons séries synchrones.

Nous disposons maintenant de deux moyens pour faire dialoguer des équipements informatiques : la liaison parallèle et la liaison série. Tous nos problèmes ne sont pas résolus pour autant. En effet, si la majorité des équipements informatiques utilisent des mots de 8 bits pour travailler, ils ne les exploitent pas de la même façon et, si l'on ne prend pas quelques précautions, nos deux appareils vont bien pouvoir échanger ces mots mais ils seront incapables de se comprendre s'ils ne leur affectent pas, tous deux, la même signification.

le code ASCII

Les informations manipulées par les micro-ordinateurs sont quelconques mais peuvent toujours être ramenées à une suite de caractères tels ceux que l'on rencontre sur le clavier de toute machine à écrire. En effet, on trouve là tout ce qu'il faut pour travailler dans de nombreux domaines puisque l'on y trouve des chiffres, des lettres et des symboles divers.

Il pourrait exister une multitude de codages différents de ces symboles mais, fort heureusement, l'un d'entre eux a été normalisé et est employé sur l'immense majorité des appareils du marché, grands ou petits, c'est le code ASCII (prononcé ASKI) qui signifie « American Standard Code for Information Interchange » ou encore code américain standard pour échange d'informations. Ce code n'utilise, dans sa version de base, que sept bits. Il peut donc représenter 128 caractères, ce qui était plus que suffisant pour toutes les applications qui existaient lors de sa création et de sa normalisation mais qui, à notre époque, présente de cruelles lacunes. Le tableau officiel de ce code ASCII, qui est, répétons-le, le seul normalisé même si certains fabricants tentent de faire croire le

contraire, vous est présenté **figure 5**. Les caractères contenus dans le cadre gras sont classiques sur tout clavier informatique et, à l'exception de quelques-uns, sur tout clavier de machine à écrire. La partie gauche de ce tableau contient les codes des caractères dits de contrôle.
Afin d'être des plus pratiques, ce tableau vous indique le codage des caractères en binaire mais aussi en hexadécimal ; ainsi A est-il égal à 1000001 en binaire ou à 41 en hexadécimal.
Comme vous pouvez le constater, il manque un certain nombre de signes ou lettres dans ce tableau dont nos est présenté **figure 6** ; code qui est bien évidemment respecté par tous les fabricants et fournisseurs de machines compatibles PC.

les liaisons séries asynchrones

Nous avons vu que la liaison série synchrone, intéressante en raison du faible nombre de fils de connexion qu'elle nécessite, devait impérativement faire appel à des horloges rigoureusement synchronisées à l'émission comme à la réception.

VALEUR DECIMALE	→	0	16	32	48	64	80	96	112
↓	VALEUR HEXA DECIMALE	0	1	2	3	4	5	6	7
0	0	blank	►	blank	0	@	P	'	p
1	1	☺	◄	!	1	A	Q	a	q
2	2	☻	↕	"	2	B	R	b	r
3	3	♥	‼	#	3	C	S	c	s
4	4	♦	π	$	4	D	T	d	t
5	5	♣	§	%	5	E	U	e	u
6	6	♠	▬	&	6	F	V	f	v
7	7	•	↨	'	7	G	W	g	w
8	8	◘	↑	(	8	H	X	h	x
9	9	○	↓	)	9	I	Y	i	y
10	A	◙	→	*	:	J	Z	j	z
11	B	♂	←	+	;	K	[	k	{
12	C	♀	∟	,	<	L	\	l	\|
13	D	♪	↔	–	=	M	]	m	}
14	E	♫	▲	.	>	N	^	n	~
15	F	☼	▼	/	?	O	_	o	Δ

VALEUR DECIMALE	→	128	144	160	176	192	208	224	240
↓	VALEUR HEXA DECIMALE	8	9	A	B	C	D	E	F
0	0	Ç	É	á	░	└	╨	α	≡
1	1	ü	æ	í	▒	┴	╤	β	±
2	2	é	Æ	ó	▓	┬	╥	Γ	≤
3	3	â	ô	ú	│	├	╙	π	≥
4	4	ä	ö	ñ	┤	─	╘	Σ	⌠
5	5	à	ò	Ñ	╡	┼	╒	σ	⌡
6	6	å	û	ª	╢	╞	╓	µ	÷
7	7	ç	ù	º	╖	╟	╫	τ	≈
8	8	ê	ÿ	¿	╕	╚	╪	Φ	°
9	9	ë	Ö	⌐	╣	╔	┘	θ	•
10	A	è	Ü	¬	║	╩	┌	Ω	•
11	B	ï	¢	½	╗	╦	█	σ	√
12	C	î	£	¼	╝	╠	▄	∞	ⁿ
13	D	ì	¥	¡	╜	═	▌	φ	²
14	E	Ä	₧	«	╛	╬	▐	∈	■
15	F	Å	ƒ	»	┐	╧	▀	∩	blank

Figure 6 : tableau du code ASCII « 8 bits » adopté par IBM

Figure 6 : Tableau du code ASCII "8 bits" adpoté par IBM.

caractères accentués (mais aussi les caractères accentués propres à d'autres langues, par exemple). Ce code a donc été étendu à 8 bits ce qui donne 128 possibilités de codage supplémentaires. Ces extensions ne sont pas normalisées, ce qui implique un certain nombre de problèmes de compatibilité que vous avez peut-être rencontré et dont nous parlerons ultérieurement.
En ce qui concerne le monde des PC et des PS, IBM a défini le code ASCII étendu dont le tableau vous

Cette condition étant souvent difficile à réaliser, un autre type de liaison série a été mis au point : la liaison série asynchrone.
Cette liaison demeure de type série et respecte donc les principes vus précédemment mais, cette fois, les circuits utilisés à l'émission et à la réception ne sont plus de simples registres à décalage. Il s'agit de boîtiers un peu plus complexes qui ont pour nom UART (« Universal Asynchronous Receiver Transmitter » ou émetteur récepteur universel de données asynchrones).
De tels UART, interfacés avec des microprocesseurs, comportent en interne, outre un registre à décalage, toute une circuiterie apte à générer les signaux particuliers d'une liaison série asynchrone. Comme leur nom l'indique, les UART contiennent un émetteur et un récepteur. Il suffit donc d'un boîtier pour établir une liaison série bidirectionnelle.
L'idée de la liaison série asynchrone reste la même que celle de la liaison série synchrone. De ce fait, il existe à l'émission et à la réception deux horloges qui doivent fonctionner à la même fréquence. Par contre, ces fréquences peuvent différer de quelques pourcentages et, surtout, les horloges n'ont plus besoin d'être synchronisées. La mise en oeuvre d'une telle liaison est donc très aisée puisqu'il suffit de réaliser, côté émission et côté réception, une horloge à quartz toute simple (il suffit pour cela d'un quartz et d'un circuit intégré). En effet, le plus mauvais des quartz assure déjà une exactitude de fréquence supérieure à 10-4.
Toute l'astuce d'une liaison série asynchrone repose sur la forme des signaux envoyés, signaux qui permet-

Figure 7 : Allure d'un caractère dans une transmission série asynchrone

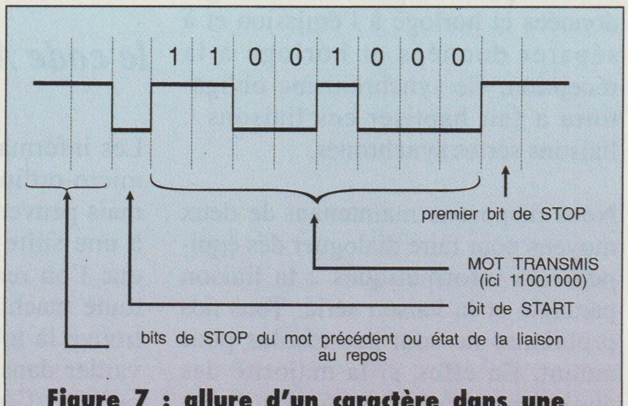


Figure 7 : allure d'un caractère dans une transmission série asynchrone

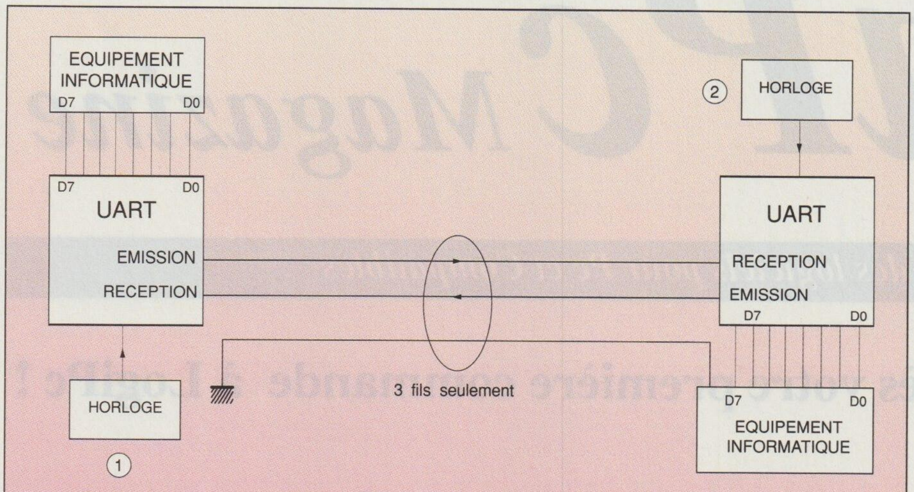


Figure 8 : une liaison série asynchrone typique

tent une resynchronisation du récepteur sur chaque caractère reçu. Examinez la **figure 7** qui représente la transmission asynchrone de l'octet « 11001000 ». Au repos, la ligne de transmission est à l'état logique haut. La transmission débute par le passage à 0 de cette ligne pendant une période de l'horloge de transmission ce qui constitue le bit de « start » (qui signifie « début » en américain). Les bits du mot à transmettre sont ensuite envoyés derrière ce bit de start comme dans une transmission série synchrone et, après le dernier bit « utile », la ligne passe à nouveau à l'état haut pendant une ou deux périodes d'horloge pour constituer ce que l'on appelle le ou les bits de stop.

En d'autre termes, chaque caractère transmis est encadré par deux bits particuliers qui en précisent le début et la fin. Ces bits sont automatiquement mis en place et enlevés par les circuits UART précités. Leur présence est donc transparente pour l'utilisateur qui, comme nous le verrons, se contente d'écrire le mot à transmettre dans un registre de l'UART en émission, tout comme il se borne à venir lire le mot reçu dans un autre registre de ce même UART en réception.

Comme le montre la **figure 8**, les horloges d'émission et de réception sont appliquées directement sur les UART (parfois même elles sont intégrées dans ces boîtiers) et c'est grâce à cela qu'une telle liaison peut fonctionner. En effet, grâce au bit de start, l'UART récepteur se synchronise sur le signal reçu et vient analyser chaque bit reçu au milieu de sa durée théorique comme le montre la **figure 9**. Il suffit alors que les fréquences des horloges émettrices et réceptrices soient assez voisines pour ne pas produire un décalage trop important sur la durée d'un mot transmis, et pour que tout se passe bien. Par ailleurs, du fait de cette resynchronisation au début de chaque caractère, les horloges émettrices et réceptrices n'ont besoin d'aucun lien entre elles et sont donc totalement asynchrones, d'où le nom donné à ce type de liaison.

Maintenant que nous avons vu les principes généraux d'une telle liaison, il nous faut préciser quelques points importants. Tout d'abord, les horloges appliquées sur les UART ne sont que très rarement égales aux fréquences de transmission mais en sont en général des multiples de 16 afin de faciliter le travail de la logique interne des circuits et, surtout, les phases de synchronisation et d'échantillonnage des bits reçus au milieu de leur durée théorique.

Les fréquences de transmission se mesurent en bauds et il existe un certain nombre de valeurs normalisées que nous allons voir dans la suite de cette série. Un baud étant égal à un bit par seconde (dans ce cas) et la transmission d'un caractère (c'est-à-dire d'un octet, si l'on utilise le code ASCII) nécessitant en général 10 bits, on parle aussi de vitesse de transmission en caractères par seconde qui est donc égale à la vitesse en bauds divisée par 10. Du 300 bauds est ainsi du 30 caractères par seconde.

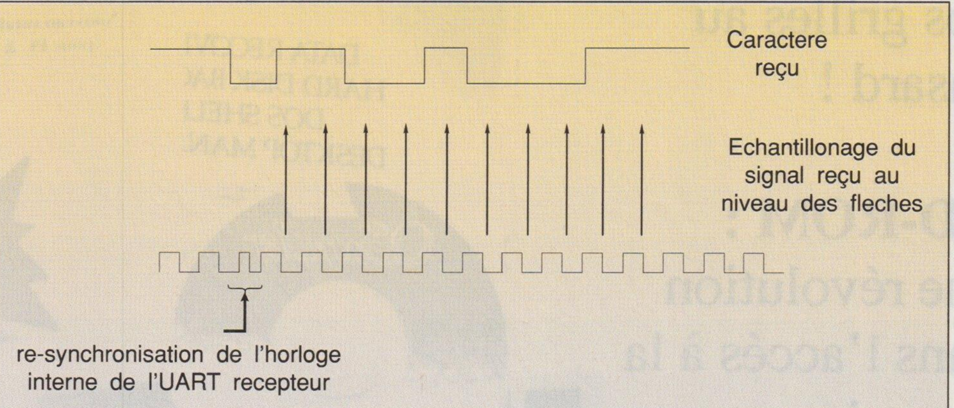


Figure 9 : échantillonage d'un caractère reçu sur une liaison série asynchrone par un UART

Nous verrons le mois prochain de nouvelles normes relatives aux liaisons séries asynchrones dont la célèbre norme RS-232, source de bien des soucis, très souvent sans fondement.

A suivre...

Christian Tavernier

LogiPc Magazine

Le bimestriel des logiciels pour Pc et Compatibles

Magazine remboursé dès votre première commande à LogiPc !

DOSSIERS:
120 logiciels du Domaine Public

SOFTLOTO:
ne cochez plus vos grilles au hasard !

CD-ROM :
une révolution dans l'accés à la connaissance.

SHOPPING :
tempête sur les prix de rentrée !



Nouveau !!! LogiPc Magazine est disponible en kiosque.

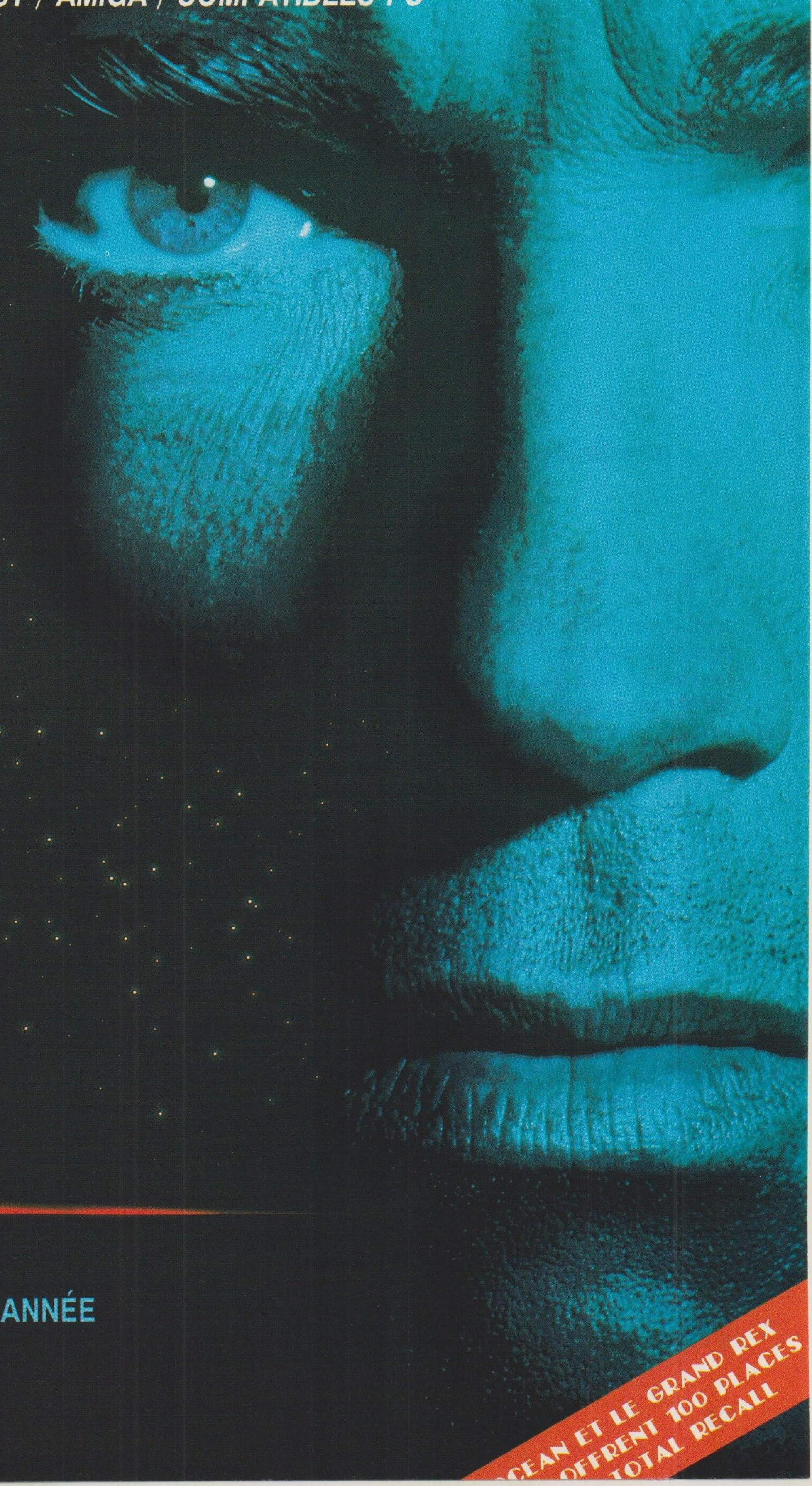
M 4408

Bimestriel - Numéro 1 - 15 septembre 1990

GÉNÉRATION 4

ATARI ST / AMIGA / COMPATIBLES PC

N°26 - Octobre 90

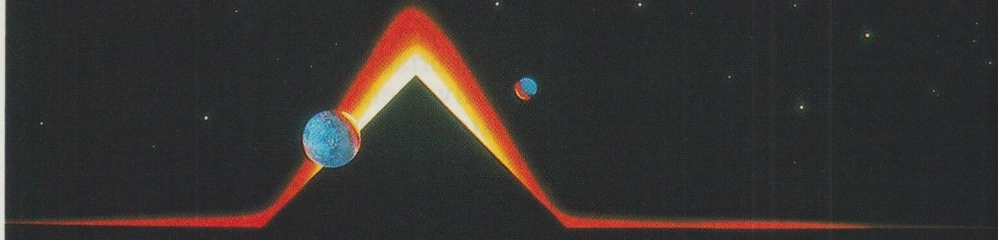


TOTAL RECALL

LE JEU LE FILM

PREVIEWS:

NARC
STRIDER 2
CHASE HQ 2
SPEEDBALL 2
TEST DRIVE 3
MIDWINTER 2
STUN RUNNER
UN SQUADRON
OFF ROAD RACER
TEENAGE HERO
MUTANT TURTLES
et bien d'autres...



SPECIAL CES DE LONDRES:
TOUS LES JEUX DE LA FIN D'ANNÉE

M 4681 - 26 - 25,00 F-RD

CEAN ET LE GRAND REX
OFFRENT 100 PLACES
TOTAL RECALL

Turbo Forth sur PC...

Langage ou Religion

Turbo Forth concilie la puissance et la compacité de l'assembleur à un haut niveau d'abstraction, avec en prime... l'exotisme.

Sans doute avez-vous déjà entendu prononcer, à mi-voix, interrompant une conversation entre amateurs de langages informatiques, le mot « Forth ». Celui qui a glissé cette parole mystérieuse a une étincelle passionnée dans le regard. Il pratique, ou a pratiqué, le Forth. Il utilisait peut-être, il y a de cela déjà quelques années, un de ces rares ordinateurs familiaux, le Jupiter Ace par exemple, possédant un Forth en lieu et place du traditionnel Basic. Ses interlocuteurs, devisant des avantages respectifs du C et de Pascal, le regarderont sans doute d'un air condescendant. Probablement, considèrent-ils le Forth comme une curiosité de laboratoire, ou encore une version mutante de casse-tête chinois informatique. Un objet étrangement beau, mais dont on a du mal à comprendre l'utilité.

Pourquoi cette réaction ?

Les érudits savent que le Forth n'est autre que le « quatrième » (amputé de son « u » qui ne rentrait pas) dans la langue de Shakespeare. Ce langage a été ainsi nommé, car il représente une « quatrième » manière d'envisager la programmation. D'après son nom, on devine également immédiatement son origine *« born in the USA »*. A l'époque de sa création, on le comparait à Fortran, Cobol et Assembleur, les langages de programmation de l'époque. Aujourd'hui, on l'opposerait plutôt à Fortran, Basic ou Lisp, Forth restant un langage d'un quatrième type.

Fortran, ou désormais Pascal, C ou Modula 2 sont les dignes représentants des langages itératifs compilés. Bien des concepts sont apparus depuis ce vénérable ancêtre, mais l'itération n'a jamais été remise en cause.

Le Basic était le prototype des langages itératifs interprétés. Depuis, les compilateurs Basic ont fleuri, et les langages purement interprétés se font rares... mais le Basic d'alors vit encore dans les mémoires.

Le Lisp est l'outil privilégié de l'intelligence artificielle.

L'utilisation systématique de la notion de fonction et de la récursivité permettent de remplacer l'itération. Suivant ses implémentations, il est plus ou moins compilé...

En le plaçant dès l'origine dans cette position « différente », les créateurs du Forth ont choisi de l'isoler. Certains non pratiquants prétendent qu'on entre en Forth comme on entre en religion... on peut difficilement leur donner tort. Pourtant, si le Forth présente effectivement des particularités passionnantes, il est beaucoup plus proche des autres langages qu'on ne pourrait le croire.

Forth et les piles

Le langage Forth est basé sur le concept de piles. Toutes les opérations sont réalisées sur ce que

l'on appelle « la pile opératoire ». Cette manière de calculer est bien connue des utilisateurs de calculatrices Hewlett-Packard. Pour faire une addition, au lieu d'écrire « 1 + 2 » comme en Basic, en Pascal, en C, ou en Fortran, on écrira « 1 2 + », c'est-à-dire que l'on précise d'abord les nombres sur lesquels va porter le calcul, puis seulement l'opérateur. Si on imagine ces nombres comme des assiettes, l'une d'entre elles représente le nombre « 2 », l'autre le nombre « 1 ». La pile opératoire est comme une pile d'assiettes : au sommet de cette pile on commence par déposer « 1 », puis « 2 ». Le mot clé « + » retire automatiquement les deux valeurs au sommet de cette pile, réalise l'opération d'addition, puis y dépose à nouveau le résultat. Si ce mode de fonctionnement peut paraître étrange à certains — pourquoi s'encombrer d'une pile alors que des langages s'en passent ? —, il ne surprendra pas ceux qui ont déjà interfacé un programme en C ou Pascal avec de l'assembleur. En effet, tous les langages évolués contemporains traduisent les expressions arithmétiques et les appels de fonctions sous cette forme (à quelques optimisations près). Forth se contente de rendre visible une structure habituellement dissimulée. Plus même, comme c'est encore cette structure de pile qui est utilisée pour effectuer les appels de fonctions et transmettre les paramètres dans tous les langages itératifs, Forth n'est pas obligé de faire la distinction entre les opérateurs « +, -, *, /, etc. » et les fonctions prédéfinies comme les opérateurs mathématiques, les appels systèmes, les impressions sur écran, etc. Les mêmes mécanismes et la même syntaxe sont systématiquement employés.

On dit souvent que le C est un langage de bas niveau — on le traite même parfois d'assembleur évolué — parce que beaucoup de ses structures internes sont visibles. Sous ce rapport, on pourrait dire de Forth qu'il est le langage de plus bas niveau qui existe : rien n'est caché, au contraire, des structures comme cette « pile opératoire » sont au cœur même du langage.

le dictionnaire

Le Forth inclut une autre structure de données importante : le « dictionnaire ». C'est dans ce dictionnaire que sont rangés tous les mots du langage. Qu'est-ce qu'un mot ? En C, en Pascal, ou même en Basic, on distingue deux sortes de mots :

- des délimiteurs syntaxiques. Les mots comme « BEGIN », « WHILE », « VAR » ou « TYPE » en Pascal ; comme « for », « int », l'adressage pointé avec l'étoile, ou encore les accolades en C, comme « IF », « THEN », « ELSE », etc., en Basic, et d'une manière générale tous les mots clés et symboles réservés du langage ;
- la deuxième sorte de mots est définie par l'utilisateur, ou bien fournie dans une bibliothèque de fonctions. Ce sont les noms des procédures, des variables, des types de données. Ces noms subsistent (sous une forme codée) même après la compilation, ils correspondent aux adresses mémoires des fonctions, des variables, etc.

... et la pile

En Forth, il n'existe pas de différence entre ces deux sortes de mots. Toutes deux sont rangées dans le dictionnaire et présentent un comportement identique. Nous avons vu comment les opérateurs travaillaient avec la pile opératoire, une étape du fonctionnement de Forth n'a pas été décrite : comment une ligne de programme est-elle interprétée ? Quand un nombre, un caractère, ou une chaîne de caractères est reconnu, il est simplement rangé au sommet de la pile opératoire. Le processus est plus complexe lorsqu'il s'agit d'un nom de fonction. Le mot est tout d'abord recherché dans le dictionnaire. Quand il est trouvé, il suffit d'exécuter le code associé. Quand il s'agit d'une variable globale et non pas d'un opérateur, au lieu d'exécuter un code, le moteur Forth se contente de renvoyer son adresse.

A ce point, il faut éclaircir quelques mécanismes. Un programme Forth n'est ni compilé, ni interprété. Il est les deux simultanément. Les commandes de compilation font partie des mots du langage : une des opérations les plus fréquentes en Forth consiste à créer un nouveau mot, c'est une phase de compilation. Compiler un mot se résume donc à interpréter une commande de compilation, avec comme paramètres un programme. On retrouve donc encore les deux catégories précédentes dans les mots clés de Forth, même s'ils sont présents dans un unique et même dictionnaire. Certains n'offrent de sens qu'à l'intérieur d'un bloc en cours de compilation, d'autres aussi bien dans une ligne à compiler qu'en mode direct. Un des intérêts immenses de cette architecture est de permettre d'enrichir facilement le langage en lui ajoutant, par exemple, des structures de contrôle.

Ces principes font de Forth un outil puissant, mais qui se programme d'une manière qui paraît très étrange au néophyte. On programme en définissant de nouveaux mots. Ceux qui connaissent les langages à objets ont pu apprécier la puissance de cette approche (bien que très différents

à d'autres égards, les langages orientés objets se programment également par ajouts de mots).

pas si différent

Nous avons examiné les points communs entre le Forth et les langages itératifs traditionnels, mais le Lisp est resté dans l'ombre. En fait, ceux qui pratiquent le Lisp ont sans doute remarqué les nombreux points communs avec le Forth. Ces deux-là, même s'ils sont de conception différente, se ressemblent énormément. La meilleure analogie qui me vienne à l'esprit est celle d'un gant.

Si le Lisp est ce gant, le Forth est le même gant retourné : on voit mieux les coutures et les mécanismes sous-jacents, et l'on ne se soucie guère de présentation ni d'interface évoluée.

Comme le Forth, on a considéré pendant très longtemps le Lisp comme un langage mystérieux. Comme le Forth, il avait et il a toujours ses adeptes passionnés. S'il a gagné ses lettres de noblesse, c'est qu'il permettait des opérations impossibles, ou très pénibles avec d'autres langages. Il a été adopté par la communauté internationale des chercheurs en intelligence artificielle. Il a servi à construire les premiers systèmes experts, à essayer des théories et des approches nouvelles de l'informatique. En deux mots, le Lisp a fait ses preuves. Enfin, s'il a été adopté en Europe, c'est beaucoup grâce à « Le_Lisp », son implémentation réalisée par l'Inria.

Forth est arrivé plus tard, pour cette raison ou par le fait du hasard, il n'a pas été choisi comme outil en I.A. Malgré des pratiquants convaincus et actifs, il lui reste encore à s'imposer. La sortie de Turbo Forth, commercialisé par Rem Corp, quel que soit son impact commercial, permet au langage Forth d'exister sur PC.

Cette implémentation sur PC dispose d'autres atouts. D'une part, c'est le plus petit langage (par la mémoire vive nécessaire) disponible sur PC : 26 ko. Avec Turbo Pascal 3.0, Turbo Forth est un des seuls langages capables de tenir dans, disons le Portfolio d'Atari, ou quelques autres portatifs de même catégorie à faible capacité de mémoire. C'est tout l'inverse de Le_Lisp qui exige plusieurs centaines de kilooctets, voire plusieurs mégaoctets, pour fonctionner convenablement. Dans ces 26 ko, les créateurs de Turbo Forth sont parvenus à offrir toute la palette des primitives désirables : des opérateurs de toutes sortes, des structures de contrôle évoluées, des fonctions graphiques, et même une gestion de multiprocessus...

un outil de développement

Malgré sa réputation, Forth est un véritable outil de développement. Il a été utilisé dans l'industrie dans des contextes assez surprenants : il s'agissait par exemple d'écrire un programme destiné à un microprocesseur de contrôle industriel. Quand, comme dans ce cas, la contrainte majeure est la mémoire, le Forth est beaucoup plus efficace que le C. Plus surprenant encore, le même programme écrit directement en Forth et en assembleur est en général plus court... en Forth ! Ce constat, qui semble incroyable à première vue, devient une évidence quand on regarde le noyau d'exécution de Forth de plus près (on peut parfaitement regarder le noyau de Turbo Forth et toutes ses primitives de très, très près... Son source, en Forth, est fourni).

des produits connus

Des produits assez connus ont été écrits en Forth. Turbo Forth est d'ailleurs accompagné de divers utilitaires, dont un clone de WordPerfect (la partie texte ASCII), en 16 Ko ! Qui dit mieux ?

Cette première version de Turbo Forth est limitée à 64 Ko pour les exécutables. Pour réaliser des programmes de taille plus importante, il faut employer la technique des « overlays ». Comme le Forth est extrêmement peu gourmand en mémoire, cette capacité est amplement suffisante pour écrire des applications de taille moyenne sous DOS. Mais mieux, dans le vocabulaire de Turbo Forth, les instructions des processeurs 8086 ont été incluses, avec une syntaxe « à la Forth ». Les opérandes en premier, puis les opérateurs. On peut donc programmer en assembleur à partir de Turbo Forth, et cela avec si peu de difficulté que j'ai vu (de mes yeux) des programmeurs sous Forth taper des programmes en assembleur directement sur la ligne de commande ! Rien d'étonnant à ce que Forth soit livré avec des programmes systèmes puissants, comme RSINIT qui permet d'émuler un minitel avec son PC.

pour les sceptiques

Forth n'est pas le remède miracle à tous les problèmes posés par l'informatique. Pour l'utiliser convenablement, il faut acquérir un mode de raisonnement très spécifique. Cela risque d'être plus simple pour un novice que pour un informaticien expérimenté. Forth plante parfois le système, non pas qu'il soit bogué, mais parce qu'il laisse aux programmeurs la possibilité de faire absolument tout. Si vous

pouviez modifier le code de votre compilateur en exécutant vos programmes, est-ce que vous vous étonneriez qu'il cesse parfois de fonctionner. Certains disent que Forth est illisible. Ce n'est pas mon point de vue : un langage de programmation est toujours illisible de prime abord. Même des programmes en Pascal, langage réputé pour sa clarté, demandent un long apprentissage avant qu'on puisse les lire facilement.

A propos d'apprentissage, l'aide en ligne de Turbo Forth est très pratique. On utilise l'opérateur « HELP » suivi d'un paramètre (le mot à expliquer), et son mode d'emploi (un texte) est affiché. On peut enrichir le mode d'emploi pour ses propres commandes en utilisant le mot « LEARN ».

ouverture à l'Est

Le Forth suit même l'actualité, puisque malgré ses défauts, Turbo Forth a été choisi comme langagage de programmation par les universités des pays de l'Est. Il a été vendu (à des conditions très intéressantes, certes) à la Yougoslavie, l'URSS, la Tchécoslovaquie... Il a été donné à la Roumanie, contre la promesse que toute application réalisée en Turbo Forth serait commercialisée par Rem Corp. Il n'est pas certain que la Roumanie ait gagné au change, mais c'est un beau geste.

Le choix du bloc de l'Est paraît judicieux. En effet, pour exploiter pleinement son parc de plus de 150 000 PC, surtout des XT, il fallait un langage puissant et peu gourmand en mémoire, et pas trop cher. De plus, choisir un langage de référence différent de ceux des pays de l'Ouest garantit d'une pénétration trop forte des produits de l'Ouest et empêche un départ massif des dévelop-



peurs de l'Est ! Les développeurs s'expatriant trop rapidement ayant quelque mal à trouver du travail à l'Ouest.

Pour le moment, Turbo Forth n'est disponible que par correspondance. Plus tard, il sera vendu par le réseau de distribution traditionnel au prix de 1 400 FTTC. En attendant, une souscription exceptionnelle a été lancée à la suite du SICOB, le produit est donc disponible pour quelques mois encore à 995 FTTC. En revanche, les documentations sont en cours de fabrication. Vous recevrez donc d'abord les disquettes, puis le « manuel 1 », qui est en français (rédigé sur un ton familier qui se laisse lire, mais qui présente le défaut de comparer sans cesse Forth et Basic, ce qui est saugrenu), enfin le manuel de référence. Pour le moment, Rem Corp n'imprime que le nombre de manuels correspondant aux produits effectivement vendus. Comme les pays de l'Est n'ont pas eu droit à une documentation en papier, on ne peut pas dire que Rem Corp ne pense pas aux arbres...

Christophe Grosjean

Toutes les applications Microsoft sur disque virtuel

Dans les numéros précédents d'Action PC, nous avions proposé des utilitaires permettant d'installer Excel et Word 5 sur disque virtuel. Ces programmes présentaient quelques imperfections, la plus grave d'entre elles étant leur incompatibilité (dans certains cas) avec la version 4 du DOS. Plutôt que d'en publier une nouvelle version, nous avons décidé de régler définitivement le problème, du moins jusqu'à ce que Microsoft abandonne ce type de protection. Voici donc en exclusivité, un installateur universel d'applications Microsoft sur disque virtuel. Cette préparation du disque virtuel n'est pas en réalité indispensable à toutes les applications Microsoft protégées. Il existe parfois dans le répertoire du logiciel un fichier d'extension « .ini » indiquant (entre autres choses) ou trouver la protection. Ce fichier est créé après la première exécution du programme. C'est le cas pour « Word » et « Word pour Windows », par exemple. Si l'on place ce fichier en plus de l'exécutable sur le lecteur de destination, le programme peut être utilisé sans préparation du disque virtuel. L'application consultera le disque dur pour vérifier les fichiers de protection. Cependant, on perd ainsi une partie de l'utilité du disque virtuel. Et si l'on souhaite placer Word (par exemple) sur le disque virtuel d'une machine connectée à un réseau, alors que Word (par exemple) est ailleurs dans le réseau, la mise en oeuvre de notre technique devient indispensable. Il faut donc éviter de recopier le fichier « ini » lorsque l'on souhaite une véritable installation sur disque virtuel. (Sachant qu'en contrepartie on perd les informations conservées dans ce fichier.) L'utilisation de l'installateur pour disque virtuel est particulièrement simple. Il suffit de savoir dans quel répertoire du disque dur l'application Microsoft à été installée. On indique le chemin d'accès à ce répertoire, ainsi que la désignation d'unité du disque virtuel à MSVIRT, puis notre utilitaire se débrouille. La syntaxe sera, par exemple, de la forme : « MSVIRT c:\winword e: » pour installer « Word pour Windows » à la racine de l'unité e:. L'installation s'effectue toujours à la racine du disque virtuel pour décourager une utilisation frauduleuse de l'utilitaire. Une fois le disque préparé, il suffit de transférer les exécutables indispensables, dans le cas de notre exemple : « copyc:\winword\winword.exe e:\ » devrait faire l'affaire. Les protections prises en compte sont celles des logiciels Microsoft récents. Il est possible que d'anciennes méthodes de protection résistent à MSVIRT. Il est également probable que MSVIRT puisse se jouer des protections d'autres éditeurs... Pour finir, un dernier avertissement, après une utilisation de MSVIRT sur un disque virtuel, utilisez « Kill » (voir Action PC n°1) pour nettoyer ce disque, ou rebootez votre PC. Surtout, évitez d'utiliser MSVIRT sur un disque dur. En effet, ces derniers s'avèrent plus difficiles à nettoyer que les disques virtuels. Ce programme a été compilé par la dernière version de Turbo C (Turbo C++), mais reste parfaitement compatible avec les versions précédentes. L'adapter au C de Microsoft ne devrait poser aucun problème, le C utilisé étant parfaitement standard. Il suffit de changer les nom de quelques fonctions, différents d'un compilateur à l'autre.

Christophe Grosjean

```
/*****************************************************/
/*              Programme MS VIRT                    */
/*      Copyrigth Action PC & C. GROSJEAN 1990       */
/*****************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <dir.h>
#include <dos.h>
#include <errno.h>
#include <string.h>
#include <io.h>
#include <process.h>
#include <dir.h>
#include <ctype.h>


char virtpath[20] = "X:\\";
char virtlg0;


/*****************************************************/
/* change d'unité de disque par appel au dos         */
/* le nom d'unité est dans une chaine de caractères  */
/* si cette chaine ne commence pas par un nom d'unité */
/* la fonction ne fait rien                          */
/*****************************************************/
void changedisk( path )
char * path;
{

        char * p;

        if (( p = strchr( path , ':')) != NULL )
        {
                setdisk( toupper(*(-p))-'A');
        }
}
/*****************************************************/
/* Transforme un chemin d'accès (relatif par exemple) */
/* en chemin d'accès complet,
                  */
/* renvoie 0 si c'est possible -1 si erreur          */
/*****************************************************/

int filtrer(path)
char * path;
{
        char courant[80];

        getcwd( courant , 80);
        if ( chdir( path ) == -1)
                return(-1);
        changedisk(path);
        getcwd( path , 80 );
        changedisk(courant);
        chdir(courant);
        return(0);
}
/*****************************************************/
/* Traitement des erreurs fatales                    */
/*****************************************************/

void erreur( message)
char * message;
{
        printf(message);
        printf("\n");
        exit(1);
}
/*****************************************************/
/*     recopie un fichier d'un directory dans un autre */
/*****************************************************/

void copy( path, virtpath )
char * path, * virtpath;
{
FILE * source, * target;
int p;
char buffer[2048];

        if ((source = fopen(path,"rb")) == NULL)
                erreur("Impossible de transfèrer la protec-
tion");
        if ((target = fopen(virtpath,"wb")) == NULL)
                erreur("Impossible d'installer la protection
sur ce ramdrive");
        do
        {
                p = fread( buffer, sizeof(char), 2048,
source);
                fwrite( buffer, sizeof(char), p , target);
        }
        while (p == 2048);
        /* on espère que ça marche */
        fclose(source);
        fclose(target);
}
/*****************************************************/
/*  fabrique un répertoire spécial sur disque virtuel */
/* copie tous les fichiers du répertoire de protection */
/*  vers le nouveau répertoire
        */
/*****************************************************/

void ms_install(path, path_attrib)
char * path, path_attrib;
{
#define FA_TOUS (FA_RDONLY | FA_HIDDEN | FA_SYSTEM | FA_DIREC |
FA_ARCH)

        struct ffblk ffblk;
        char t;
        char lg, virtlg;

        if (mkdir(virtpath)== -1)
                erreur("Erreur en accès au disque virtuel");
        else
        {
                strcat(path,"\\");
                lg = strlen(path);
                strcat(virtpath,"\\");
                virtlg = strlen(virtpath);
                {
                        char generique[80];

strcat(stpcpy(generique,path),"*.*");
```

```
                        t = findfirst( generique, &ffblk,
FA_TOUS);
                }
                while (t == 0)
                {
                        if (ffblk.ff_name[0] != '.')
                        {
                                strcat(path,ffblk.ff_name);
strcat(virtpath,ffblk.ff_name);
                                if   (_chmod(path,   1,
FA_ARCH) == -1)
                                        erreur(“Erreur en
accès à la protection”);
                                copy(path, virtpath);
                                _chmod(path,1,FA_RDONLY |
FA_HIDDEN);
                                _chmod(virtpath,1,FA_RDONLY
| FA_HIDDEN);
                                path[lg] = 0;
                                virtpath[virtlg] = 0;
                        }
                        t = findnext(&ffblk);
                }
                path[lg-1] = 0;
                virtpath[virtlg-1] = 0;
                if ( _chmod(virtpath, 1,FA_RDONLY | FA_HID-
DEN) == -1)
                 printf(“attention : attributs du répertoire
de protection incorrects\n”);
        }

}

/*************************************************/
/* reconnait les noms de fichiers                */
/* ou de répertoires spéciaux                    */
/* des protections Microsoft                     */
/* (contenant un <espace> avant d'autres caractères) */
/*************************************************/

char special(filename)
char *filename;
{
        char i;

        for (i = 0; ((i<8) && (filename[i] != 32)); i++);
        for( i = ++i ; i<8 ; i++)
          if (filename[i] != 32) return 1;
        return 0;
}
/*************************************************/
/*         recherche un directory MS,            */
/*      puis le copie sur disque virtuel         */
/*************************************************/

char show(path)
char * path;
{
#define FA_TOUS (FA_RDONLY | FA_HIDDEN | FA_SYSTEM | FA_DIREC |
FA_ARCH)

        struct ffblk ffblk;
        char t, fait;
        char lg = strlen(path);

        if (path[lg - 1] != '\\')
        {
                (strcat(path,"\\"));
                lg++;
        }
        {
                char generique[80];

                strcat(stpcpy(generique,path),"*.*");
                t = findfirst( generique, &ffblk, FA_TOUS);
        }
        fait = 0;
        while (( t == 0 ) && (fait == 0))
        {
                /* on reste dans le répertoire indiqué sans parcourir
d'arborescence */
                if (special(ffblk.ff_name))
                        {
                                s t r c a t ( p a t h ,
ffblk.ff_name);
strcat(virtpath,ffblk.ff_name);
                                if   (ffblk.ff_attrib   &
FA_DIREC)
                                        ms_install(  path  ,
ffblk.ff_attrib);
                                else
                                if   (ffblk.ff_attrib   &
```

```
FA_LABEL)
                                        fait—; /* c'est le nom de
volume, pas une protection !*/
                                else
                                {
                                        if  (_chmod(path,
1, FA_ARCH) == -1)
erreur(“Erreur en accès à la protection”);
                                        copy(path,  virt-
path);
                                        /*  a  ce  niveau
reste à espèrer que ça a marché */

_chmod(path,1,FA_RDONLY | FA_HIDDEN);
                                        if  (_chmod(virt-
path,1,FA_RDONLY | FA_HIDDEN) == -1)
                                                printf(“attention
: attributs du fichier de protection incorrects\n”);
                                        path[lg] = 0;
                                }
                                virtpath[virtlg0] = 0;
                                fait++;  /* une seule pro-
tection, pour les appli Microsoft */
                        }
                        t=findnext(&ffblk);
        }
        path[lg-1] = 0;
        return fait;
}

/************************************************************/
/* analyse de la ligne de commande, messages d'erreurs */
/* appels de filtrer et de show                        */
/************************************************************/

void main( argc , argv )
char *argv[];
{
        char path[80],i , nbpath = 0;

        for  ( i = 1 ; i < argc ; i++)
        {
                if ( nbpath == 0)
                {
                        nbpath++;
                        strcpy(path,argv[i]);
                        if ( filtrer(path) == -1)
                                printf(“Répertoire %s inac-
cessible\n”, path);
                                exit(1);
                        }
                }
                else if (nbpath == 1)
                {
                        nbpath++;
                        if ( isalpha(argv[i][0])
                         && (argv[i][1] == ':')
                         && (argv[i][2] == 0))
                        {
                                virtpath[0] = argv[i][0];
                                virtlg0 = strlen(virtpath);
                        }
                        else
                                erreur(“Spécification de
disque virtuel incorrecte”);
                }
                else
                        printf( “Trop d'arguments\n”);
        }
        if (nbpath != 2)
        {
                printf(“MSVIRT  V1.0  :  Copyright  1990
Christophe Grosjean\n”);
                printf(“ prépare le disque virtuel pour une
application Microsoft protégée\n”);
                printf(“usage     : MSVIRT <répertoire> <ram
drive>\n”);
                printf(“  <répertoire>  :  répertoire  de
l'application\n”);
                printf(“  <ram drive>     désignation d'unité
du disque virtuel\n”);
                printf(“exemple : MSVIRT c:\\winword e:\n”);
                exit(1);
        }
        if (show(path))
                printf(“Préparation du disque virtuel termi-
née\n”);
        else
                printf(“Pas de protection détectée dans le
répertoire %s\n”, path);
}
i
```

Qu'est-ce que...

LES SERVEURS SQL

Après la présentation, dans le dernier numéro, de la syntaxe de base de SQL, étudions de plus près la structure des serveurs SQL.

Classiquement, un serveur SQL est un moteur de bases de données, capable de réaliser des relations logiques sur les données. Ce type de logiciel, installé sur un site central puissant, doit être capable de stocker les données et de traiter des requêtes multiples en provenance des différentes stations de travail connectées par l'intermédiaire d'un réseau. Ce sont actuellement les serveurs SQL, correspondant au modèle client/serveur, qui font figure de standard sur le marché des SGBD (pour Systèmes de Gestion de Bases de Données) devant supporter plusieurs dizaines d'accès simultanés.

L'avantage fondamental du modèle client/serveur sur un réseau de PC est dû au fait que seules circulent sur le réseau les requêtes d'interrogation et les résultats issus de ces requêtes. Les algorithmes de traitement et les résultats intermédiaires sont stockés localement sur le serveur et n'occasionnent pas ainsi de transport d'informations inutiles.

On pourrait penser que tous les serveurs SQL disponibles sur le marché fournissent les mêmes fonctionnalités. En réalité, il n'en est rien. Chaque DBMS (pour Data Base Manager System) présente des caractéristiques différentes pouvant le rendre plus ou moins adapté à une application donnée.

Certains serveurs SQL, en plus des commandes standard, offrent des outils supplémentaires pour le traitement des chaînes de caractères, des fonctions mathématiques et statistiques, des nouveaux types de données, etc. Certains serveurs, tels que Ingres ou InterBase, permettent également de construire ses propres fonctions.

Il est donc très important, avant d'arrêter un choix dans ce domaine, de bien définir ce que sera l'application pour trouver le

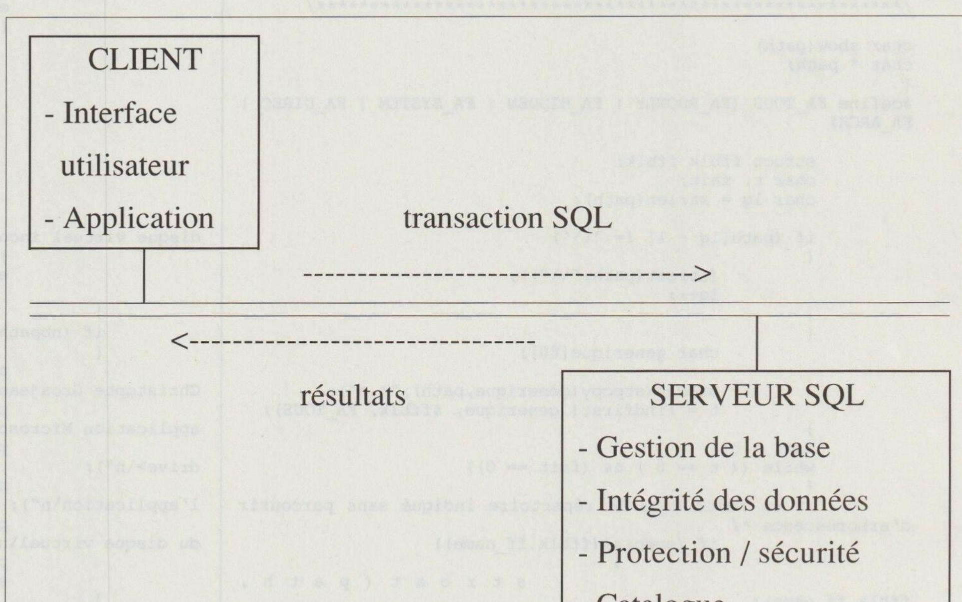


Schéma d'un serveur SQL sur une architecture réseau

serveur qui répondra le mieux à ces besoins spécifiques. Il est également crucial de pouvoir juger de la pérennité des outils choisis : c'est-à-dire du serveur employé, du système d'exploitation (OS/2, Unix, VAX/VMS, etc.), et du réseau local (3Com, Novell, IBM—Tokeng-Ring, etc.) qui supportera l'application. Enfin, les capacités du serveur à passer d'une plate — forme système à une autre constituent également un élément déterminant dont il faudra tenir compte.

et si l'on parlait sécurité...

Sur un serveur SQL, le gestionnaire de transactions préserve l'intégrité des tables de données et des index - et ceci même après une erreur de programme ou une erreur système. Dans le cas d'un « plantage », si une transaction n'est pas totalement terminée (un COMMIT non achevé par exemple), le gestionnaire de transactions doit supprimer les mises à jour entamées et revenir à l'état initial (avant le COMMIT) des tables et index.

Les éventuelles erreurs disques ne doivent pas non plus influer sur la base de données. Ce cloisonnement est obtenu grâce à des utilitaires de sauvegarde (backup/restore) et des procédures d'enregistrement des transactions réalisées et achevées depuis la dernière sauvegarde.

Un serveur SQL assure d'une part l'intégrité d'entité — chaque enregistrement est unique dans une table — et d'autre part l'intégrité référentielle. Ainsi, imaginons l'exemple d'une application de gestion de commandes, avec une table fournisseurs et une table commandes. Il ne doit pas être possible de passer une commande à un fournisseur n'existant pas dans la table fournisseurs. Ce contrôle intelligent est assuré par le DBMS, qui stocke les règles d'intégrité au niveau du catalogue de la base. Le développeur pourra également créer ses propres règles d'intégrité à l'aide des déclencheurs (ou triggers) : il s'agit de procédures disponibles sur le serveur et automatiquement activées au cours de certaines opérations (création, modification, destruction, etc.). Tous les serveurs SQL ne sont néanmoins pas équivalents en ce qui concerne ces fonctionnalités. Si aucun de ces mécanismes (intégrité d'entité, intégrité référentielle, déclencheurs) n'est implémenté au niveau du serveur, c'est lors de la phase d'écriture de l'application proprement dite que le programmeur devra envisager les problèmes d'intégrité.

des serveurs dévoués

La plupart des serveurs SQL peuvent être intégrés à des langages hôtes tels que le C, Cobol, Pascal, ADA, etc. Font exception à la règle des produits tels que Progress, ne comprenant que leur propre L4G (langage de quatrième génération).
Tous les serveurs présentent une architecture B—tree qui permet un accès séquentiel ou direct aux données. Ils doivent bien sûr être capables de gérer les accès concurrentiels à une même information, tout en empêchant un éventuel blocage né d'une interférence en cours de transaction. Si un tel « interblocage » apparaît, l'une des transactions est momentanément abandonnée (elle sera relancée ultérieurement). Un « Rollback » est activé, permettant la libération de la ressource nécessaire à l'autre transaction.

Après cette approche théorique, dressons un état du marché des serveurs SQL.

SQL SERVER :

C'est un peu le standard du modèle client/serveur. Il tourne sous Unix, OS/2, VAX/VMS, et la version OS/2 fonctionne sur les réseaux Novell ou LAN Manager (3Com, IBM LAN Server, etc.).

INFORMIX :

C'est le plus célèbre des serveurs dans le monde Unix. Il fonctionne également sous VAX/VMS. L'architecture est reprise par Informix-Star.

ORACLE :

Oracle propose différentes plates-formes : Unix, OS/2, VAX/VMS. Le produit SQL-Star offre des possibilités de répartition sur plusieurs serveurs. Une passerelle existe vers DB2, SGBD d'IBM tournant sur de très gros systèmes. Notons l'absence de contrôle de l'intégrité référentielle par le serveur.

INGRES :

Il offre dès maintenant des possibilités d'architecture multiserveur. Ingres tourne sous Unix et VAX/VMS, et bénéficie d'un excellent interpréteur de commandes SQL.

PROGRESS :

Réservé aux adeptes des L4G. Le SQL de Progress ne peut s'intégrer à un langage hôte de type Pascal, C, ou ADA. Il existe des versions pour Unix et VAX/VMS.

Un peu de prospective...

L'avenir des bases de données peut s'envisager sous le signe de la répartition et des bases de connaissances. La répartition de la base (sur un ensemble d'ordinateurs hétérogènes connectés en réseau — on dépasse alors le stade du serveur unique contenant l'ensemble de la base et seul capable de fournir une information) nécessite une transparence physique absolue vis-à-vis de l'utilisateur, qui ne doit pas avoir à se

Mini glossaire

accès direct :
accès aux enregistrements par l'intermédiaire d'un index qui permet de retrouver directement l'emplacement d'un enregistrement dans le fichier.

accès séquentiel :
accès aux enregistrements dans l'ordre dans lequel ils ont été saisis, en commençant par le premier.

architecture multiserveur :
architecture répartie, où plusieurs systèmes informatiques sont dédiés au stockage des données et au traitement des requêtes.

base de données répartie :
la base de données ne se trouve plus centralisée sur un site unique, mais elle est éclatée sur un ensemble d'ordinateurs reliés entre eux par un réseau physique. L'accès transparent aux données par les utilisateurs devient alors le but principal à atteindre.

base de connaissances :
la base contient non seulement un ensemble de données, mais également des règles sur ces données.

catalogue :
base de données système contenant des informations sur les tables de base, les vues, les droits d'accès, l'identification des utilisateurs, etc. Sa constitution varie d'un système à l'autre.

Commit :
commande SQL qui rend permanents les changements effectués sur la base de données, au cours d'une transaction.

déclencheur :
procédure stockée sur le serveur, activée automatiquement lors d'une requête, qui permet de renforcer la cohérence des données.

interblocage :
un interblocage (ou verrou mortel) apparaît lorsqu'une transaction cherche à obtenir une ressource déjà verrouillée par une autre. Elles se bloquent donc mutuellement et ne peuvent s'achever.

serveur :
unité matérielle et logicielle devant stocker, traiter et extraire des données en les rendant disponibles pour des stations de travail au sein d'un réseau.

station de travail :
tout système informatique comprenant une unité de calcul, des périphériques de saisie (écran, clavier, souris, tablette à digitaliser, etc.), éventuellement des périphériques de stockage (lecteur de disquettes, disque dur, bande, etc.), et des dispositifs de communication, dédié à un traitement par un utilisateur unique.

transaction :
suite d'opérations telle que chaque opération de cette suite est nécessaire pour atteindre un résultat unitaire. Une transaction se termine par un « COMMIT » ou un « ROLLBACK ». Elle est donc soit complètement exécutée, soit complètement abandonnée.

verrouillage :
processus de restriction de l'accès à une ressource dans un environnement multitâche.

Rollback :
commande SQL qui supprime les changements effectués sur la base de données au cours d'une transaction. Cette commande s'invoque après une panne système.

soucier de la localisation des données. Des mécanismes d'appels de procédures à distance (RPC pour Remote Procedure Control) doivent être implémentés. Ce type de mécanismes permet à un serveur de déclencher des procédures résidantes sur un autre serveur, offrant entre autres la possibilité d'établir des transactions parallèles.

Une base de connaissances, ensemble de données et de règles sur ces données, peut être utilisée pour la construction de systèmes experts. Les interrogations sont maintenant réalisées non seulement sur les données contenues dans la base, mais aussi sur les conclusions déduites à partir des données et des règles. Un problème nouveau se pose pour ce type de bases : il devient en effet très difficile de contrôler la mise à jour et l'ajout de nouvelles règles, de préserver la cohérence avec les règles existantes, et d'éviter les redondances.

En vue de ces futurs développements, les bases de données relationnelles, et plus particulièrement celles basées sur le modèle SQL, semblent en mesure de répondre à ces nouveaux besoins exprimés.

Pascal Fievet

PcFCX
Pour compresser tous types de fichiers et réduire leur encombrement. FCX gère les multi-volumes si une disquette ne suffit pas, sait compresser et restituer les arborescences de répertoires et supporte les caractères de remplacement * et ?.
Fonctionne sous MS-DOS et VAX/VMS.
Prix version MS-DOS 801,00 ht

TERM
Logiciel de communication asynchrone avec langage de description de scripts. Fonctionne sur plus de 60 systèmes différents, MS-DOS, Mac, Unix, Xenix, etc... avec émulation Minitel.
Prix version MS-DOS 2200,00 ht

RDM
Puissant générateur d'applications complètes avec base de données et macro langage intégrés. Fonctionne sur MS-DOS, Xenix et VAX/VMS, les applications sont portables entre tous ces systèmes.
Prix version MS-DOS 6900,00 ht

WordPerfect
Traitement de texte avec fonctions PAO intégrées.
Prix version MS-DOS 4500,00 ht

Bon de Commande

Produit	Qté	Prix Unit. ht	Prix Tot. ht
Total			
TVA 18,6 %			
Total TTC			

Adresse de livraison

Livraison standard sur disquettes 5'1/4.
Facture jointe à tous nos envois franco de port.

ENOSYS
105 rue des Prés aux Bois
78220 VIROFLAY
Tél (1) 30 84 45 41 - Télécopie (1) 30 84 43 10

Cerclez info pub 29

SÉCURITÉ INFORMATIQUE

RESCUE est un outil de diagnostic logiciel dont la vocation est d'assurer une pré-maintenance chez l'utilisateur final.

Ses nombreux utilitaires lui permettent :
- ✓ de diagnostiquer de nombreuses applications spécifiques et standard (Lotus 123, DBase III, Works, Windows, MS-DOS...) (1)
- ✓ de reconnaître plus de 500 types de fichiers
- ✓ d'identifier plus de 70 virus

Principales anomalies détectées :
- ✓ modification ou perte de fichiers
- ✓ erreurs de manipulation logicielle et/ou matérielle
- ✓ intrusions extérieures (virus...)

Toute anomalie est aussitôt signalée par l'apparition à l'écran d'un message rédigé en langage clair.

RESCUE est livré avec un manuel didactique sur les virus et les anti-virus.
RESCUE fonctionne sur ordinateurs : IBM PC, XT, AT, PS et compatibles.

(1) Lotus 123, DBase III et Works, Windows, MS-DOS sont des marques déposées respectivement par Lotus International, Asthon Tate et Microsoft.

LA SOLUTION *LOG'UP*

Rescue 1.00

Bon de commande

A retourner à :
LOG'UP S.A. 22, rue de Metz l'Evêque
80000 - Amiens Tél.: 22-91-05-05

❑ Mme ❑ M :
Fonction :
Entreprise :
Adresse :
Code Postal : Ville :
Tél. : Fax :

souhaite
❑ recevoir un exemplaire de RESCUE au prix de 1400 francs H.T. (soit 1660,40 Frs TTC). Pour une commande de plusieurs exemplaires, nous consulter.
❑ recevoir de plus amples informations sur le logiciel RESCUE et ses utilitaires intégrés.

Ci-joint règlement par :
❑ chèque bancaire ❑ CCP
❑ carte bleue n° :
Signature :

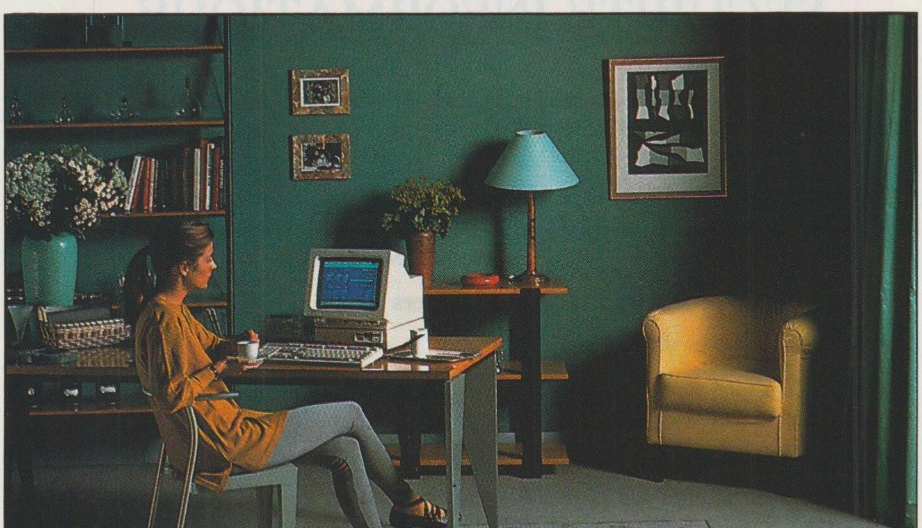
Les chèques doivent être libellés au nom de LOG'UP S.A.

Cerclez info pub 30

IBM PS/1

UN IBM A 9 000 FRANCS

IBM attaque le marché de la micro-informatique domestique avec le PS/1. Le premier d'une série de produits destinés à bouleverser le marché.



Le minitel, présent dans presque tous les foyers français, n'effraie plus personne. C'est un outil pratique, utilisé tant pour des applications professionnelles que comme moyen de distraction. IBM, et plusieurs autres grands constructeurs, pensent que l'ordinateur domestique peut jouer un rôle identique. L'idée n'est pas neuve, mais aujourd'hui la technologie est mûre, et les prix suffisamment bas pour permettre de proposer un matériel réellement exploitable au niveau professionnel. Le premier à s'engager, IBM, peut s'accaparer le marché avec le PS/1. Le produit est compact, assez puissant, efficace avec des applications professionnelle simples (professions libérales, commerçants, etc.) et doté d'un caractère ludique très marqué (graphismes, musique, etc.).

Il faut simplement ne pas perdre de vue qu'il s'agit d'un ordinateur domestique, impossible à comparer avec un véritable ordinateur professionnel (tant au niveau du prix que des performances).

Précisément, le petit dernier d'IBM est un AT (80286) à bas prix, destiné au grand public. Il constitue le nouveau point d'entrée de la gamme micro-informatique d'IBM. Quatre configurations sont disponibles. Elles possèdent en commun un processeur 80286 à 10 MHz, un lecteur de disquettes 3"1/2 de 1,44 Mo, un écran VGA, un clavier AZERTY de 102 touches, une souris IBM avec deux boutons, des sorties série, parallèle et souris, un haut parleur avec réglage du volume et prise pour casque d'écoute, un modem miniature à la

norme V23 (compatible Hayes) qui permet de se connecter au réseau minitel. Le modem se branche sur le port série et ressemble à une grosse prise. Du côté logiciel, on retrouve le système d'exploitation DOS 4.01, le logiciel Works 2.0 de Microsoft, un logiciel d'émulation minitel, et un logiciel spécifique de gestion du PS/1. La documentation se compose d'un livre pour la mise en marche, d'un guide d'utilisation, d'un mémento, du manuel de références de Microsoft Works ainsi que de son aide-mémoire. Les quatre configurations portent les noms très imagés (comme de tradition chez IBM) de modèle 101, modèle 201, modèle 134 et modèle 234. Le modèle 101 constitue la version de base. Il possède 512 Ko de mémoire vive et un écran VGA monochrome. Le Modèle 201 comporte aussi 512 Ko de mémoire mais l'écran est en couleur. Le modèle 134 est livré avec 1 Mo de mémoire vive, un disque dur de 30 Mo, et un écran VGA monochrome. Enfin, le modèle 234, le plus complet, comprend 1 Mo de mémoire, un disque dur de 30 Mo et l'écran VGA couleur. Les prix sont fixés à 9 200 FTTC pour le modèle 101, 12 600 FTTC pour le modèle 201, 14 200 FTTC pour le modèle 134 et 17 000 FTTC pour le modèle 234 (prix relevés à la Fnac de La Défense, IBM ne communiquant pas de prix public conseillé).

Deux remarques s'imposent : les modèles 101 et 201, de par l'absence de disque dur, restent un peu limités. Cependant, on peut faire fonctionner — dans des conditions acceptables — l'intégré Works fourni avec l'appareil. Rappelons que cet excellent logiciel propose en fait quatre logiciels en un : traitement de texte, tableur, gestion de fichiers et communication. Ces quatre applications englobent 90 % du marché de la micro-informatique professionnelle. Et, si Works ne peut rivaliser avec quatre logiciels spécialisés (par exemple, le traitement de texte Word, le tableur Excel, le gestionnaire de données Paradox et le logiciel de communication DynaComm) dont la valeur totale peut atteindre 20 000 F, il se montre à son avantage dans des applications professionnelles simples. A tel point que certains utilisateurs avertis emploient Works 2.0 pour des besoins spécifiques. Works comprend par ailleurs le meilleur didacticiel (module d'apprentissage) du marché et une aide en ligne.

La seconde remarque concerne l'écran monochrome du PS/1, compatible avec la norme VGA. Cela signifie que tout logiciel prévu pour fonctionner avec des écrans VGA couleur tournera sur les PS/1 monochromes, les couleurs étant transformées en niveaux de gris. Cela n'était pas vrai, par exemple, avec les moniteurs Hercules qui ne permettaient pas l'emploi de logiciels conçus pour des écrans couleur CGA ou EGA. La couleur, dans le cas du PS/1, n'est nullement indispensable. Elle apporte simplement un confort visuel supplémentaire, notamment pour les jeux et les logiciels éducatifs. Dans le cas d'une utilisation exclusivement professionnelle, le choix d'un PS/1 avec moniteur monochrome paraît plus judicieux, et représente une économie non négligeable (environ 3 000 F).

de nombreuses extensions

Avant de décrire les autres logiciels livrés en standard, détaillons les extensions disponibles. La première option est une carte permettant d'augmenter la mémoire des modèles 101 et 201 de 512 Ko de mémoire à 1 Mo. Un disque dur de 30 Mo est aussi disponible pour ces deux modèles. C'est un point important, l'acheteur pouvant démarrer avec un système PS/1 à un prix très attractif, et le faire évoluer — en protégeant son investissement — en fonction de ses besoins. Si un écran couleur est également disponible pour les modèles 101 et 134, on prendra soin de se renseigner auprès du distributeur s'il reprend l'écran monochrome, sous peine de se retrouver avec un écran devenu inutile sur les bras. Notons que l'achat du moniteur IBM est obligatoire, l'alimentation de l'ordinateur se trouvant dans l'écran. Ce choix conduit à un gain de place important, mais limite un peu l'évolution de l'affichage (les Amstradistes ont déjà connu ce désagrément avec la norme CGA des PC 1512). Cependant, la norme VGA est une des plus performantes actuellement, ce qui réduit les risques. Les prix de ces extensions ne sont pas connus car, ne servant qu'à l'évolution d'un modèle vers un autre, elles ne sont pas disponibles au moment de la rédaction de ces lignes.

On trouve également deux lecteurs de disquettes proposés en option : le deuxième lecteurs de disquette 3"1/2 de 1,44 Mo et un lecteur de disquettes de 5"1/4 de 1,2 Mo (2 500 FTTC).

Par ailleurs, les amateurs d'informatique ludique seront heureux d'apprendre qu'une carte audio (1200 F) peut être directement insérée dans un port spécial du PS/1. Cette carte comporte un échantillonneur polyphonique à quatre voies, une prise pour manette de jeux, un connecteur pour un microphone et une interface Midi. Il existe aussi une version plus complète de cette carte (1500 F) qui dispose, en outre, d'un boîtier Midi comportant les ports « in », « out » et « through » et d'un câble d'interface MIDI. De plus, cette carte est livrée avec une manette de jeux, et un ludiciel (un jeu d'arcade de qualité médiocre, « Silpheed », dont l'intérêt premier est de démontrer les possibilités sonores de la carte musicale). La carte audio n'est malheureusement pas compatible avec les cartes déjà existantes (Ad-Lib ou MT-32), ce qui oblige les éditeurs à récrire leurs logiciels de jeux pour l'exploiter. Les nouveaux jeux, par contre, ne devraient pas poser de problème. La société Fretless International propose déjà des logiciels, entre autres un séquenceur, exploitant la carte

audio des PS/1, et un câble pour raccorder la carte à un clavier ou à un piano Midi. Une deuxième manette de jeux est aussi disponible (environ 400 F). Le PS/1 ne comportant pas de connecteur d'extension, une unité externe (1500 FTTC) comprenant deux slots de 16 bits et un slot de 8 bits permet d'ajouter trois cartes d'extension. Voilà qui est significatif de la philosophie du PS/1 : alors que commercialiser un PC professionnel sans connecteur constituerait une hérésie, on se demande bien à quoi pourrait servir de tels connecteurs sur le PS/1. En version de base, tous les périphériques nécessaires en environnement domestique sont disponibles (carte graphique, souris, modem, manette de jeu, carte musicale, etc.). On n'imagine pas de relier le PS/1 à un réseau, ni de lui adjoindre un scanner ou un streamer. Des besoins souvent incontournables en entreprise.

quatre logiciels en standard

Quatre logiciels sont livrés en standard avec le PS/1. En premier lieu le système d'exploitation DOS 4.01 dont une partie est définitivement enregistrée en mémoire morte (ROM). Ceci permet d'exploiter un PS/1 avec un seul lecteur de disquettes sans jouer au « grille-pain » avec la disquette du DOS et la disquette d'application. Le second logiciel, Works 2.0 de Microsoft, a été précédemment évoqué. Il est livré sur disquettes pour les modèles 101 et 201, et directement installé sur le disque dur pour les modèles 134 et 234. Dommage qu'IBM ne fournisse pas les disquettes de Works pour les modèles à disque dur, l'acheteur ne disposant d'aucun moyen pour réinstaller le logiciel en cas de perte accidentelle. Seul le revendeur pour résoudre un éventuel problème. Cela empêche aussi de reconfigurer son système, par exemple, en partitionnant le disque. Même s'il est vrai qu'en environnement domestique, ce type de besoin est rare. Le troisième logiciel, « Overcom », une émulation minitel (en effet, Works étant d'origine américaine, son module de communication ne sait pas gérer notre minitel national). Ce logiciel, ainsi que le modem du PS/1 sont des produits de GOTO Informatique. Le quatrième logiciel, développé spécialement pour le PS/1 (entièrement piloté par la souris) affiche l'écran graphique d'accueil et ses quatre grosses icones. La première permet d'accéder à l'aide de Works ou à celle du système PS/1. Cette dernière décrit les différentes composantes du système par l'intermédiaire d'images. La deuxième icone lance directement Works. La troisième démarre un logiciel de gestion des disques (shell). Grâce à un système graphique, l'utilisateur se déplace dans l'arborescence du disque et exécute les programmes qu'il contient. La quatrième icone transmet le contrôle au DOS.

un serveur spécifique IBM˜PS/1

L'émulation minitel et le modem donnent accès au serveur d'IBM : 3615 IBMPS. Ce serveur, réservé au PS/1, livre le diagnostic rapide d'une panne éventuelle ou d'une mauvaise manipulation de l'ordinateur. Il fonctionne 24 heures sur 24 (même le jour le Noël, comme l'a souligné non sans malice un dirigeant d'IBM). Il donne aussi des informations sur le PS/1 pour les acheteurs potentiels : domaines d'utilisation, descriptif commercial, annonces, etc. Il fournit également la liste des distributeurs agréés PS/1 et des centres de réparation IBM. Une messagerie assure un dialogue permanent entre les utilisateurs et le service après-vente d'IBM. Ce serveur est pour l'instant entièrement dédié au PS/1, mais IBM prévoit de l'étendre au PS/2. Il s'agit d'une excellente initiative qui ne laisse aucun utilisateur démuni, quelles que soient sa localisation géographique et ses heures de travail sur PS/1.

un seul carton et cinq minutes

Derrière ce serveur se cache un système expert développé spécialement par une société française. En plus du serveur minitel, un point réponse est aussi au service des utilisateurs et des acheteurs potentiels. Le numéro vert 05.03.03.03 donne des informations complémentaires sur le produit.

Le PS/1 est livré dans un unique carton (sauf le modem et le logiciel d'émulation minitel, livrés dans une boîte séparée). Son assemblage est des plus simple, comparable au branchement d'un magnétoscope sur un téléviseur. Une prise de courant et une prise de téléphone suffisent pour que le PS/1 soit complètement opérationnel. Lorsque l'on allume l'ordinateur, on accède aussitôt à l'écran d'accueil. Aucun logiciel ne doit être installé ni chargé. On retrouve ici les éléments de la campagne de publicité d'IBM : « Cinq minutes et vous êtes dans le coup. »

pour imprimer, il faut une autre prise de courant

Parallèlement à la sortie du PS/1, IBM présente une imprimante personnelle référencée 2205. Il s'agit d'une imprimante matricielle à 24 aiguilles, bidirectionnelle, de 80 colonnes, pourvue d'une mémoire tampon (buffer) de 5 000 octets. Elle possède deux modes de fonctionnement dédiés : qualité listing avec une vitesse de 192 cps (caractères par seconde), et qualité courrier avec une vitesse maximum de 53 cps. Quatre polices sont disponibles : Sans Sérif, Courier, Prestige et Gothic. La résolution graphique peut atteindre 180 points par pouce verticalement, et 360 points par

pouce horizontalement. Il est possible d'utiliser du papier en continu (listing) ou en feuille à feuille. Une fonction de mise en attente du papier en continu permet d'imprimer quelques feuilles libres sans être obligé de retirer puis de réinstaller le papier listing. L'imprimante comporte aussi une barre de découpe pour le papier en continu. En option, on trouve un chargeur feuille à feuille automatique d'une capacité d'environ cent feuilles et une option tracteur pour le papier en continu. Cette dernière permet d'éviter d'éventuels bourrages de papier (le papier listing est poussé).

Le marché visé par IBM se décompose en trois segments. Le premier est constitué de professionnels. Il comprend les travailleurs indépendants, les artisans et commerçants, les professions libérales et les travailleurs à domicile. Une configuration monochrome avec disque dur paraît la mieux adaptée pour ce groupe. Le second segment est celui du marché familial qui semble connaître une nouvelle jeunesse, notamment avec les consoles Nintendo. Les jeux (avec la carte audio, une manette et l'écran couleur, on obtient une console de jeu d'une qualité exceptionnelle), la formation personnelle et l'éducatif nécessitent un écran couleur mais le disque dur n'est pas indispensable. Le dernier segment est celui de l'enseignement (les étudiants, professeurs et chercheurs, les lycées, les grandes écoles, et les universités. Si ce marché est plus particulièrement développé aux Etats-Unis, (où l'antique Apple II reste assez présent), IBM pourrait s'intéresser au besoins de l'éducation nationale où quelques milliers (sic) de Thomson obsolètes devront être remplacés.

et on le trouve où ce petit ?

Le PS/1 n'est pas au catalogue d'IBM mais est disponible auprès

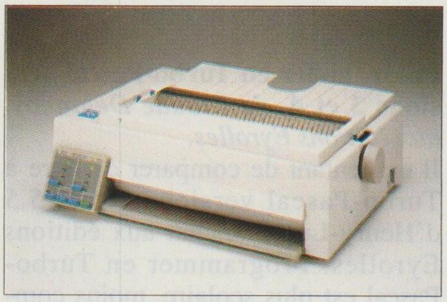


des revendeurs agréés PS/1. Pour une plus large diffusion, à peu près cent cinquante nouveaux revendeurs PS/1 apparaîtront sur la France. Ces nouveaux magasins ne seront agréés que pour la vente du PS/1. Il s'agit de chaînes de distribution comme Darty, de magasins spécialisés comme la Fnac, etc.

Le prix de la machine reste très bas. Seul le prix du matériel taïwanais reste comparable, même si encore une fois, on ne peut vraiment comparer les prix ou les performances d'un IBM PS/1 avec ceux d'un Compaq Deskpro 286N. A titre d'exemple, pour situer l'offre d'IBM, on obtient, à configuration égale, un prix de 18 500 F pour l'IBM avec l'option connecteur et 15 110 F pour le compatible taïwanais en comptant un Works à 2500 F et en ajoutant 1000 F pour un modem avec l'émulation minitel. Les machines de Compaq et Tandon sont nettement plus chères mais offrent des possibilités professionnelles d'un tout autre ordre.

l'avenir en rose

Seules les possibilités d'extension du PS/1 paraissent un peu insuffisantes sur un marché du PC en évolution permanente. Cependant, si l'on compare le PS/1 à ses véritables concurrents, l'Atari ST et le



Commodore Amiga, on s'aperçoit que pour un prix certes plus élevé, il offre des performances infiniment supérieures, une architecture ouverte propre aux PC (même si elle est moins développée que dans le cas de PC professionnels), beaucoup plus de logiciels et trois lettres qui résument la force commerciale et la pérennité de son constructeur.

Cela étant, en utilisation purement professionnelle, on trouve des offres supérieures à celle d'IBM : meilleures performances chez Compaq, meilleurs prix avec les clones compatibles, meilleur rapport prix/performances chez Tandon. Si l'on prend la plus grosse configuration du PS/1, le disque dur de 30 Mo paraît léger en regard de la croissance exponentielle de la taille des logiciels PC. Cela peut poser des problèmes, même pour des étudiants. Par exemple, Turbo C de Borland dont la version 2.0 occupait 1,9 Mo, voit sa taille passer à 6 Mo pour la version la plus récente. Le Basic PDS de Microsoft peut occuper jusqu'à 14 Mo. Enfin, le passage à OS/2 est définitivement bloqué : OS/2 réclame au minimum un disque dur de 60 Mo. De même, 1 Mo de mémoire vive semble juste. IBM prévoit bien de commercialiser une extension à 2 Mo pour les logiciels gourmands, mais est-ce suffisant quand certains logiciels refusent de démarrer avec moins de 4 Mo de mémoire ! Sans parler de la version sans disque dur, qui risque de connaître des difficultés avec de nombreux logiciels qui ne peuvent même plus être exécutés depuis la disquette. Avec un PC professionnel, ces problèmes n'existent pas, mais ces machines sont plus difficiles à utiliser, leurs possibilités sonores sont nulles, et leur adjoindre une manette de jeu constitue un défi que bien peu d'utilisateurs ont relevé (y compris vos serviteurs).

Vincent Pouilley
& Eric von Ascheberg

Tous les livres sur...

LE PASCAL

Le Turbo-Pascal n'est pas mort. Non seulement Borland annonce une nouvelle version de son langage fétiche, mais le marché de l'édition suit également. Nous vous proposons ici une sélection de livres classés par thèmes.

Initiation et perfectionnement

Découvrez Turbo-Pascal sur PC ou compatibles, *par John Colibri, aux éditions Mnémodyne, deux tomes, (368 et 484 p., 158 F le tome).*

Colibri est au Pascal ce que Norton peut représenter pour le PC : un des rares auteurs qui maîtrise tellement bien son sujet qu'il parvient à l'expliquer clairement aux autres. Ceux qui programment en Pascal savent d'ailleurs que ses livres constituent une référence. Ceux qui souhaitent apprendre le Pascal ne sont malheureusement pas toujours au courant. Pourtant, il a écrit au moins quinze livres sur le sujet, dont cinq sur la version « Turbo » du langage. Découvrez Turbo-Pascal sur PC ou compatibles est donc le premier d'une longue série. Il introduit bien sûr tous les concepts de base de la programmation structurée en général, et du Turbo-Pascal versions 4.0 et suivantes en particulier. Le second tome va beaucoup plus loin je doute fort qu'il existe d'autres livres d'introduction à un langage qui se terminent par une présentation et une première mise en oeuvre des arbres binaires. Auparavant, le lecteur aura appris à concevoir une application en Pascal, à utiliser les records, les tableaux, les ensembles, les fichiers, et les pointeurs (70 pages leur sont consacrées). Les techniques les plus complexes, comme la récursivité, sont abordées et expliquées de manière simple.

Il faut rendre hommage à l'auteur sur ce point : la simplicité de ses explications et les schémas abondants dont il fait usage, associés aux nombreux programmes d'exemples toujours commentés — rachètent en grande partie la pauvreté de la typographie des Éditions Mnémodyne.

Turbo-Pascal version 4, *de Jacques Boisgontier, aux éditions Cédic / Nathan.*

Destiné aux débutants, ce livre fait un peu double emploi avec la documentation de Borland. Son intérêt est d'être clair, d'une structure nette permettant d'accéder rapidement au sujet désiré : type de fonction, procédure et fonction, graphisme, fichier. Bien qu'utile aux débutants, l'ouvrage est loin d'être exhaustif et ne s'avère pas indispensable.

Programmer en Turbo-Pascal versions 3 et 4, *de Claude Delannoy, aux éditions Eyrolles.*

Il est tentant de comparer ce livre à Turbo-Pascal versions 5.0 et 5.5 d'Henry-Luc Planchat aux éditions Eyrolles. Programmer en Turbo-Pascal est plus scolaire, moins complet, et moins clair. Beaucoup moins illustré d'exemples que son rival, il présente plus le Pascal standard (si cela a encore un sens) que le Turbo-Pascal. De surcroît, il comporte 70 pages de plus, et n'a pas des qualités de présentation suffisantes pour le faire préférer. S'il faut choisir, inutile d'hésiter.



Exercices en Turbo-Pascal, *de Claude Delannoy, aux éditions Eyrolles.*

Des exercices accompagnent le cours de Pascal présenté dans Programmer en Turbo-Pascal versions 3 et 4 (toujours la même approche scolaire). Ces exercices,

particulièrement classiques, ne sont pas dénués d'intérêt. Ils permettront de réviser par exemple les tris, la gestion de mémoire dynamique, ou la récursivité.

Programmer en Turbo-Pascal versions 4 et 5, *de Frédéric Blanc et Pierre Brandeis, aux éditions P.S.I. (295 pages, 175 francs).*
Si ce n'est pas déjà fait, allez lire la critique du Livre de Turbo-Pascal sur PC et compatibles, des mêmes auteurs chez le même éditeur. La principale différence entre les deux ouvrages tient dans le titre. Il ne s'agit que d'un enrichissement du précédent permettant d'englober la version 5.0. Le contenu a été un peu amélioré, mais ne vaut toujours pas le détour. Sans grand intérêt.

Turbo-Pascal 5 par la pratique, *par Douglas Hergert, aux éditions Sybex (622 pages, 328 F).*
Si, après avoir lu ce livre, vous n'avez toujours pas compris comment programmer une gestion de fichiers, c'est à désespérer.
Écrit pour les débutants en Turbo-Pascal 5.0, ce livre aborde les principaux aspects de la programmation dans ce langage avec force exemples. Mais Douglas Hergert semble véritablement obsédé par la notion de fichier, à tel point que toute structure de données offerte par le Turbo-Pascal se trouve réduite à cette utilisation : même les pointeurs, pourtant pas spécifiquement utiles en ce qui concerne la gestion de fichiers (tant que l'on n'aborde pas les applications séquentielles indexées lourdes), sont étudiés dans cette optique. On regrette également le peu de lisibilité des programmes, et leur disposition bizarre dans le livre : ainsi trouve-t-on un paragraphe interrompu par dix pages de programmes. Comme le paragraphe introduit le programme, on se retrouve vite perdu...
En revanche, l'introduction au langage et aux types de données est bonne. Dans l'ensemble, on peut mieux faire, et cela a été fait, mais chez d'autres éditeurs que Sybex et pour moins de 328 F.

Maîtriser Turbo-Pascal 5.5 et la programmation orientée objet, *par Claude Delannoy, aux éditions Eyrolles (400 pages, 300 F).*
Ce livre est de qualité. Peut-être vaut-il mieux avoir déjà lu Programmer en Turbo-Pascal, du même auteur, si l'on est débutant. En revanche, si le lecteur a déjà programmé, il trouvera ici toutes les informations dont il a besoin pour bien posséder le langage. Tout y est expliqué, jusqu'aux subtiles distinctions qui s'établissent pour le compilateur (mais pas forcément pour le programmeur) au moment de la déclaration de variables. On regrette cependant que les concepts quelque peu avancés, comme les pointeurs ou la récursivité, soient traités en vitesse : 7 pages sur la récursivité, 14 sur les pointeurs, c'est un peu court ! La typographie employée pour les programmes et les figures laisse également à désirer.
Enfin, la partie consacrée à la programmation objet ne joue le rôle que d'une vague initiation, sans commune mesure avec le contenu du reste de l'ouvrage. Mais on apprécie les exercices en fin de chapitre, qui permettent, à ceux qui souhaitent vraiment apprendre, de se mettre en situation, d'autant qu'ils sont corrigés (ce n'est pas le cas de tous les livres).

Turbo-Pascal, bibliothèques de procédures, *par Thomas et Jean-Édouard Lanchand-Robert, aux éditions Sybex (600 pages, 288 F).*
Remis à jour pour les versions 4.0, 5.0, et 5.5, ce livre au contenu inégal fournit au programmeur de nombreuses unités comportant des routines qui peuvent se révéler utiles. Il fait également office d'introduction à certaines techniques qui, sans être véritablement avancées, déroutent parfois le programmeur qui a dépassé le stade de débutant mais n'est pas encore confirmé. On apprendra ainsi comment interfacer Turbo-Pascal et assembleur, comment sauvegarder un écran, appeler les fonctions du DOS et du Bios, utiliser les pointeurs, etc.
Les thèmes proposés concernent essentiellement, et c'est la faille de l'ouvrage, l'affichage. A la fin du livre, on n'aura aucun mal à écrire un menu déroulant, programmer des fenêtres en mode texte, ou utiliser la souris en mode graphique.
Par ailleurs, l'expérience prouve que les utilisateurs ne modifient pas les programmes dont ils disposent (quand il les lisent !), mais qu'ils tentent de les faire coïncider avec leurs propres besoins. Une démarche évidemment mauvaise pour l'apprentissage de la programmation, mais très répandue. Sachant cela, les auteurs auraient tout intérêt à montrer comment modifier les programmes donnés, et à expliquer le plus possible leurs principes de fonctionnement. Particulièrement en ce qui concerne l'affichage, chacun faisant en ce domaine ce qui lui plaît, ou ce qu'il réussit à programmer.
Il reste toutefois une bonne introduction aux techniques de bases de l'affichage, même si la mise en pratique est discutable.

Approche de la programmation avec Turbo-Pascal, *par A. Credi, aux éditions Belin, coll. DIA, (120 pages).*
La collection DIA (éditions Belin) est supposée fournir des supports de cours aux élèves des classes préparatoires et des universités. Le moins que l'on puisse dire, c'est que l'ouvrage consacré au Pascal est décevant. En 120 pages, A. Credi a trouvé le moyen de ne fournir que trois programmes complets, dont un concernant l'affichage graphique en Turbo-Pascal version 3.0. Heureusement que les étudiants en savent plus sur la programmation que certains de leurs professeurs !

Bibliothèque de programmes en Turbo-Pascal version 4.0, *par Kris*

Jamsa et Steven Nameroff, aux éditions Borland / McGraw-Hill, traduction de Suzanne Dellinger, (468 pages, 250 F).
Kris Jamsa et Steven Nameroff écrivent souvent ensemble chez McGraw-Hill. Ils n'ont donc pas de difficulté à se lire. L'utilisateur, lui, aimerait bien pouvoir en dire autant. C'est dommage, car s'il s'agit du principal défaut de ce livre, il fait naître tous les autres. Les caractères ont incontestablement été mal choisis, et l'oeil a du mal à s'adapter lorsqu'il passe du texte au programme ou au schéma. La présentation, la typographie, l'absurde choix de mêler morceaux de programmes et texte, la liaison inextricable des programmes les uns avec les autres, qui fait qu'une ligne sur deux contient un appel à une procédure définie cent pages auparavant : autant d'éléments qui rendent ce livre absolument illisible. Pourtant, les techniques de programmation présentées sont les meilleures qui soient, le contenu « rédactionnel » est clair et progressif, et — chose rare — les programmes donnés en exemple sont pertinents.

Le Grand Livre du Turbo-Pascal, versions 5.0 et 5.5, *par Rosenbaum et Scholles, édité par Data Becker, publié par Micro-Application (720 pages, 345 F).*
Les éditions Micro-Application publient aussi un livre sur le sujet dans la collection « Bien débuter », et un autre dans leur collection « Auto-formation ». Celui-ci réussit, en plus de 700 pages, a en apprendre autant que les autres en 120 ou 200 pages.
Même si c'est le livre des grands débutants, il paraît abusif d'occuper 477 pages à expliquer ce que sont les instructions, les procédures et les unités, avant de bâcler trois programmes en 30 pages (avec les commentaires), et de reprendre ensuite des explications sur Tinst, le débogueur, et les principes de la POO. Dommage que le contenu soit vraiment trop délayé car la présentation est agréable (on note la présence de très belles reproductions d'écran).

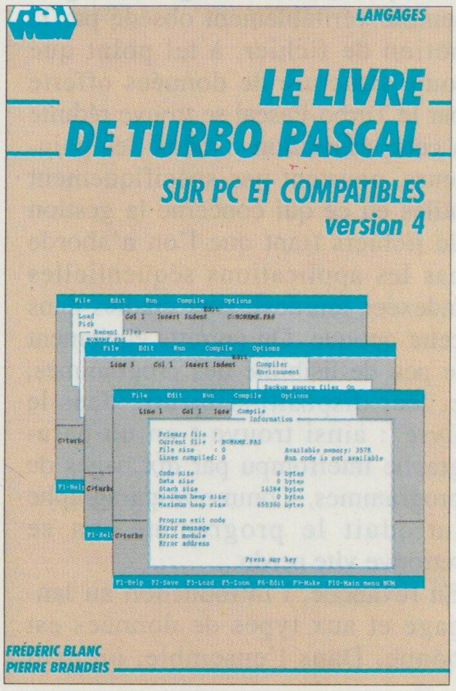
Turbo-Pascal, versions 5.0 et 5.5, *par Henri-Luc Planchat, aux éditions Eyrolles, (338 pages, 260 F).*
Il est dommage que les caractères et la mise en page de cet ouvrage soient peu agréables et gênent la lecture. Car au niveau du contenu, il s'agit d'une bonne initiation — assez poussée — à Turbo-Pascal. Toutes les notions de base sont présentées avec quelques plus : des chapitres traitent du son, des graphismes, ou de l'interface avec le DOS, voire des overlays.



Le Livre de Turbo-Pascal sur PC et compatibles (version 4), *de Frédéric Blanc et Pierre Brandeis, aux éditions P.S.I. (293 pages, 150 F).*
A peine une initiation, ce livre ne fait même pas double emploi avec la documentation de Turbo-Pascal 4. Après 293 pages, on en apprend à peu près autant avec l'adresse de l'imprimeur qu'avec le reste de l'ouvrage. A éviter.

Programmation système

Au coeur de Turbo-Pascal, et Le Désassembleur 8086 Colibri, *par John Colibri, aux éditions Mnémodynes (512 pages, 195 F - Au coeur de Turbo-Pascal), et (280 pages, 490 F - Le Désassembleur 8086, avec la disquette).*
Voici deux livres dépassés mais traitant réellement de la programmation système en Turbo-Pascal.
Après avoir écrit un assembleur en Pascal, il était logique que John Colibri écrive — toujours en Pascal —un désassembleur. Le livre contient un manuel de référence, une analyse des principes de fonctionnement du désassembleur, une explication des structures de données utilisées pour l'écrire, et le texte des programmes sources, malheureusement en Turbo-Pascal version 3.0. Depuis la disparition effective des fichiers .COM, le désassembleur servira essentiellement à l'examen de la mémoire, et surtout à comprendre comment ce genre d'utilitaire fonctionne. Avec les commentaires fournis et la disponibilité des programmes, rien n'empêche le lecteur averti d'en écrire un pour les fichiers .EXE.
Au coeur de Turbo-Pascal constitue — toujours pour la version 3.0 — une excellente étude du compilateur de Borland. On y apprend tout sur la gestion de la mémoire employée par Turbo-Pascal, l'implémentation des données type par type, le code généré, et le contenu de la librairie. Le livre se termine avec des propositions de modi-



ACTION PC LE MAGAZINE DE LA PRODUCTIVITE AMELIOREE NE RATEZ PAS NOTRE NUMERO SPECIAL PORTABLES

ALLIANCE INFORMATIQUE

Dans le cadre de son développement Alliance Informatique recrute:

- **Ingénieurs commerciaux**
- **Ingénieurs technico-commerciaux**
- **Directeurs de magasins**
- **Techniciens de maintenance**

Votre profil:
- Bac + 2
- Motivé
- une expérience en informatique
- une expérience de la vente

Nous vous offrons:
- un salaire motivant
- un poste à responsabilité
- des possibilités d'évolution importantes au sein d'une entreprise dynamique en pleine expansion.
- la possibilité d'être actionnaire dans votre propre structure...

Interessé ?

Envoyez CV + photo à
Alliance Informatique
Direction du personnel
BP 105
75749 Paris Cédex 15
(Réponse assurée)

PS: veuillez préciser le poste pour lequel vous postulez.

Cerclez info pub 38

Vous utilisez dBASE* ?

Gagnez du temps !

10 minutes = 3 heures

Ne programmez plus...

Gagnez du temps: dBTURBO, l'étonnant Turbo Générateur Automatique de programmes dBASE* écrit des centaines de lignes de manière automatique, sans aucune programmation.

Reliez automatiquement vos écrans et vos différentes bases. Faites des allers-retours d'une base à l'autre, instantanément et sans limitations.

Etonnez votre entourage: le néophyte utilise toutes les fonctionnalités de dBTURBO en moins d'une heure.

dBTURBO fonctionne sur tout système IBM Pc ou compatible, même sans disque dur. Les sources générées nécessitent pour être exécutées: dBASE III Plus*, dBASE IV*, dBFAST*, Nantucket*, ou un interpréteur ou compilateur compatible à 100%.

Prix de lancement: 2995 Frs HT, soit 3552 Frs TTC franco de port. **Manuel détaillé en Français.**

```
* Programme généré par dBTURBO
* Il peut être distribué librement
SET STATUS OFF
SET DELETED ON
SET TALK OFF
USE VOITUR INDEX VOITUR
GOTO TOP
TYPE VOITUR.MAS
...

 IF LEFT(MMARQUE,1)=' "
   DO CHARGE
   DO MSAY
   LOOP
 ENDIF
FIND&MMARQUE
 IF .NOT. EOF() .AND. MTEMPPP<>MM
  DO CHARGE
  DO MSAY
 ELSE

@ 5, 65+I GET MCOULEUR
@ 6, 29+I GET MSORTIE

@ 7, 29+I GET MCONSTR
READ
DOCONSTR.PRH
ENDDO
@ 9, 29+I GET MCYL PICTURE"9999"
@ 9, 53+I GET MTURBO
@ 10, 29+I GET MPUIDIN PICTURE"9"
DRAP='N'
.../...
USE VOITUR INDEX VOITUR
  GOTO ENREGIS
  DO REPLACE
 ENDIF
ENDIF
.../...
PROCEDURE PUBLIC
*****************
PUBLIC MMARQUE, MCOULEUR, MSO,
MCYL, MTURBO, MPUIDIN, MPUIFISC,
MLONG, MLARG, MBOITE, MCARBU
RETURN
PROCEDURE SAY
*************
@ 5, 29+I SAY MARQUE
@ 5, 65+I SAY COULEUR
@ 6, 29+I SAY SORTIE
@ 7, 29+I SAY CONSTR
@ 9, 29+I SAY CYL PICTURE "9999"
@ 9, 53+I SAY TURBO
@ 10, 29+I SAY PUIDIN PICTURE "999"
@ 11, 29+I SAY PUIFISC PICTURE "999"
```

Le Turbo Générateur qui s'amortit en 3 jours

EN CADEAU GRATUIT si vous commandez sous 8 jours: **dBGENIAL, logiciel d'interrogation et de recherches multi-critères, complément indispensable d'une valeur de 1017 Frs**

Vite, retournez ce bon à:

ABACORE, 76 rue Maréchal Joffre, 06000 NICE
Tél 93 88 48 84, Fax 93 88 14 70, **3615** code **T3S**

☐ **OUI, adressez-moi sous 24 heures mon dBTURBO et mon cadeau dBGENIAL. Je joins un chèque de 3552 F ttc**

☐ **Je désire recevoir la version limitée, avec le manuel original. Ci-joint un chèque de 150 F ttc, remboursé lors de l'achat.**

☐ **Je désire recevoir une documentation gratuite, sans engagement.**
(A remplir en MAJUSCULES. Ajouter 40 francs pour l'Etranger)

SOCIETE:

NOM et Prénom:

Adresse:

Code: **Ville:**

Téléphone: **Disquette 5"1/4** ☐ **3"1/2** ☐

APc

Cerclez info pub 31

*dBASE III+, dBASE IV: marques déposées par Ashton Tate, Pc par IBM
Nantucket et dBFAST: Marques déposées.

fication du Turbo-Pascal. Chaque élément utilisé fait l'objet d'un court programme dont on nous montre le désassemblage après compilation. S'il est évidemment devenu obsolète, ce livre permet de se familiariser avec l'assembleur, les principes de fonctionnement d'un compilateur, et la programmation système en général. A vrai dire, après avoir lu cet ouvrage, on n'a plus qu'une seule envie : le voir réécrit pour la version 5.5.

La Bible du Turbo-Pascal 5.0 et 5.5, *par Michaël Tischer, traduction de Christophe Stehly. Edité par Data Becker, publié par Micro-Application (814 pages, 340 F sans disquette, 440 F avec disquette).*
Après la Bible PC, voici la Bible Pascal. Si, comme toujours, l'auteur cède à ses préférences pour la programmation du clavier et de l'écran, et ne livre rien - ou presque - concernant les disques et disquettes, il faut toutefois reconnaître que cette bible recèle de grandes qualités. Tous les moyens mis en oeuvre pour accroître l'efficacité du Turbo-Pascal — mixage avec de l'Assembleur, appels au BIOS et au DOS, etc. — sont présentés. Dans le même ordre d'idées, c'est à notre connaissance le seul livre qui présente des programmes multitâches en Pascal. On regrette quand même une présentation désagréable (toutes les lignes de programme se terminent par un signe « <Ÿ »), des caractères trop petits, et un texte besogneux. En revanche, les programmes sont intéressants, souvent bien écrits, et tournent vraiment. A acheter (440 F) pour la disquette et le chapitre sur le multitâche : le reste se trouve déjà dans la Bible PC.

Programmation avancée en Turbo-Pascal, *par Henri-Luc Planchat, aux éditions Eyrolles (238 pages, 250 F).*
Mêmes remarques que pour le Livre Turbo-Pascal, versions 5.0 et 5.5. C'est également le même auteur (Henri-Luc Planchat) et le même éditeur (Eyrolles). C'est dommage, car le contenu a l'air intéressant. Il est question de registres, de ports, d'interface avec l'assembleur, et de souris.

Programmation Objet

Turbo-Pascal 5.5 et la programmation objet, *par John Colibri, aux éditions Mnémodynes (175 pages, 158 F).*
Depuis maintenant une année, les livres sur la programmation objet se font légion. Parmi eux, deux sujets prédominent : le C++ et le Turbo-Pascal 5.5. Les livres sur ce dernier ont toutefois l'inconvénient de se faire particulièrement discrets sur la notion de fichiers d'objets. On sait en effet que les objets sont logés en mémoire sous forme de pointeurs, et qu'il est pour cette raison (entre autres, le polymorphisme et l'héritage posant aussi des problèmes) difficile de les stocker sur disque. John Colibri s'est attelé à la tâche et propose en 180 pages une bonne initiation aux objets et à leurs fichiers. La nouvelle version de son livre, qui vient d'être publiée, approfondit encore plus le sujet.
C'est également un des meilleurs livres sur la programmation objet. Une des principales raisons tient à ce qu'il présente le code assembleur généré par les nouvelles instructions du langage ainsi que les programmes Pascal de désassemblage.

Turbo-Pascal orienté objets, *par H. Paul Raiduk, traduit par Suzanne Dellinger, aux éditions Borland-McGraw-Hill, (208 pages, 160 F).*
Cette fois encore, Borland et McGraw-Hill se sont associés pour éditer ce petit livre (200 pages), écrit par H. Paul Haiduk et traduit par Suzanne Dellinger. Quoique tout ne soit pas encore parfait, on sent une volonté d'amélioration de la présentation des livres parus dans cette collection. Celui-ci est à la fois lisible et de qualité.
En quatre chapitres avec exercices et problèmes, dont les solutions sont données en fin d'ouvrage, le lecteur s'initiera à la nouvelle approche de la programmation que constitue la POO. Le premier chapitre peut être sauté sans vergogne : il introduit au Turbo-Pascal. Le second, en revanche, est très enrichissant. La notion de type abstrait de données est intelligemment expliquée, ce qui prouve soit des qualités de pédagogie de la part de l'auteur, soit que l'ouvrage a été très bien traduit. En effet, les types abstraits de données sont généralement mal enseignés quand ils ne sont pas carrément ignorés par la plupart des ouvrages portant sur la programmation et les types de données. Or il s'agit d'une notion primordiale, assez bien adaptée aux objets, puisqu'elle permet de définir un type de données (Tableau, Enregistrement, etc.) par son « comportement ». Ainsi une pile permet au moins les opérations d'empilement (push) et de dépilement (pop). A partir de là, il est aisé de passer au concept d'objet, puisqu'un objet réunit le code (les comportements) et les données (caractérisées par un type). C'est ce qu'explique le chapitre trois, avec de nombreux exemples. Le dernier chapitre joue malheureusement un peu le rôle de fourre-tout. Y sont abordés en vrac les méthodes dites virtuelles, le polymorphisme, et la généricité. L'auteur semble avoir complètement oublié de parler des fichiers d'objets — la grande difficulté de la programmation objet. Dommage...

En vrac

Data Structures Using Pascal, *par Tennenbaum et Augenstéin, aux éditions Prentice-Hall, (767 pages, 310 F).*
C'est le livre sur les types de données. Et qui dit types de données, dit bien entendu algorithmes... Il est d'ailleurs cité dans de nom-

breuses bibliographies d'ouvrages de référence. Après avoir défini les divers types de données du Pascal standard en signalant à chaque fois les différences existant entre les principaux compilateurs et en expliquant de quelle manière ils sont implémentés, Tenenbaum examine toutes les autres structures de données utiles en informatique (comme les piles, les listes, les files, les arbres et les graphes). Il fournit à chaque fois un type de données abstrait, propose des exemples d'implémentation mettant en oeuvre des techniques diverses (tableaux et pointeurs pour les piles, par exemple), puis présente des algorithmes employant ces structures. C'est ainsi que le lecteur est initié à la récursivité, aux tris, aux recherches dans des arbres ou à l'intérieur d'une table de hachage, et même à la gestion dynamique de la mémoire.

L'auteur ne s'en tient cependant pas là : sans quoi, il ne s'agirait jamais que d'un exposé d'algorithmes comme tant d'autres. Il donne des exemples, fournit des exercices, reprend plusieurs fois le même problème en l'abordant différemment, et, surtout, il reste toujours simple à lire.

B-Arbre et séquentiel indexé, par John Colibri, *aux éditions Mnémodyne (672 pages— dont 421 pages de programmes —, 283 F).*
Ce livre, bien que présentant des programmes en Turbo-Pascal 3.0, possède de nombreuses qualités pour ceux qui s'intéressent à la programmation des bases de données séquentielles indexées. A condition d'avoir déjà de bonnes notions du langage Pascal et de la programmation en général, on ne sera pas déçu de l'ouvrage. En plus de 600 pages, le lecteur passe de la construction d'arbres binaires à celle de B-Arbres virtuels pour gérer les index de ses bases de données.

Il apprendra aussi bien à utiliser de puissants algorithmes de tri et de recherche qu'à évaluer les performances de ses programmes et à déterminer quelle est la méthode optimale à employer en fonction de ses besoins. Outre les courts programmes d'exemples présentés à l'intérieur de chaque chapitre, le lecteur dispose de 420 pages de programmes. Sont présentés un correcteur orthographique, une comptabilité (journal, grand livre, balance des comptes, consultation), une base de données à requêtes multicritères.

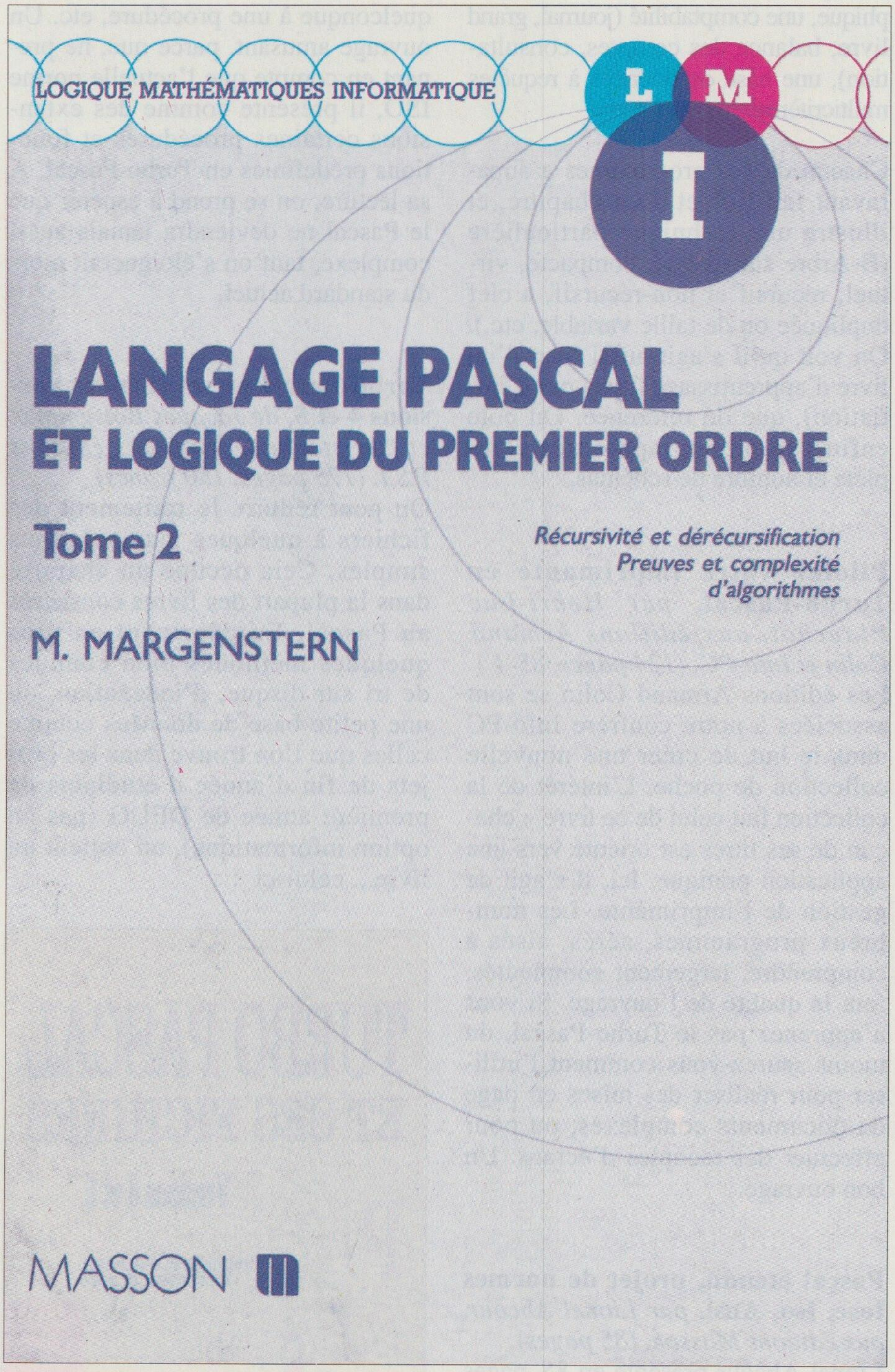
Chacun de ces programmes a auparavant fait l'objet d'un chapitre, et illustre une technique particulière (B-Arbre sur disque, compacté, virtuel, récursif et non-récursif, à clef dupliquée ou de taille variable, etc.). On voit qu'il s'agit aussi bien d'un livre d'apprentissage (mais pas d'initiation), que de référence. On note enfin une bibliographie très complète et nombre de schémas.

Pilotez votre imprimante en Turbo-Pascal, *par Henri-Luc Planchat, aux éditions Armand Colin et Info-PC, (124 pages, 85 F).*
Les éditions Armand Colin se sont associées à notre confrère Info-PC dans le but de créer une nouvelle collection de poche. L'intérêt de la collection fait celui de ce livre : chacun de ses titres est orienté vers une application pratique. Ici, il s'agit de gestion de l'imprimante. Les nombreux programmes, aérés, aisés à comprendre, largement commentés, font la qualité de l'ouvrage. Si vous n'apprenez pas le Turbo-Pascal, du moins saurez-vous comment l'utiliser pour réaliser des mises en page de documents complexes, ou pour effectuer des recopies d'écrans. Un bon ouvrage.

Pascal étendu, projet de normes Ieee, Iso, Ansi, *par Lionel Abcour, aux éditions Masson, (85 pages).*
Lionel Abcour présente en 85 pages une future norme du Pascal, dont la principale amélioration serait la prise en compte de tableaux dynamiques, et une modularité accrue. On note également un type complexe pour les nombres complexes, les procédures SeekWrite, SeekRead, et SeekUpdate de gestion de fichiers (qui effectuent un Seek et une opération de lecture, d'écriture ou de mise à jour en un seul appel), la possibilité de passer un tableau de taille quelconque à une procédure, etc. Un ouvrage amusant, parce que, ne prenant en compte que l'actuelle norme ISO, il présente comme des extensions certaines procédures et fonctions prédéfinies en Turbo-Pascal. A sa lecture, on se prend à espérer que le Pascal ne deviendra jamais aussi complexe, tant on s'éloignerait alors du standard actuel.

Turbo-Pascal et ses fichiers versions 4 et 5, *de Jacques Boisgontier et Christophe Donay, aux éditions P.S.I. (175 pages, 150 francs)*
On peut réduire le traitement des fichiers à quelques manipulations simples. Cela occupe un chapitre dans la plupart des livres consacrés au Pascal. En décrivant en plus quelques méthodes bien connues de tri sur disque, d'indexation, ou une petite base de données comme celles que l'on trouve dans les projets de fin d'année d'étudiants de première année de DEUG (pas en option informatique), on obtient un livre... celui-ci !





Langage Pascal et logique du premier ordre (deux tomes), *de M. Margenstern, aux éditions Masson.*
Certains distinguent l'informatique telle qu'elle est couramment pratiquée de la Science Informatique (avec un grand S et un grand I). Une attitude trop fréquente dans les universités est de croire que l'Informatique (avec un grand I) n'est qu'une branche des mathématiques dérivant de la logique formelle. Les deux tomes de trois cents pages tapés à la machine qui résultent de cette conception sont incompréhensibles pour le commun des mortels, et pénibles pour les malheureux étudiants qui y sont confrontés par obligation.
Le Pascal est construit à partir de sa syntaxe à l'intérieur du système formel de la logique du premier ordre.

Cela permet diverses démonstrations, sur la dérécursivation ou l'étude de complexité des programmes, par exemple. Ces démonstrations sont jolies, mais les thèmes abordés très superficiels et sans intérêt pour la programmation : l'ésotérisme n'excuse en rien l'inutilité, au contraire !

Graphisme avec Turbo-Pascal et la Graphix Toolbox, *de André Warusfel et Thomas Lachand-Robert, aux éditions Sybex.*
Malgré son titre, ce livre n'est pas exclusivement réservé aux utilisateurs de la version 3 de Turbo-Pascal (pour laquelle fut conçue la graphique ToolBox). Au contraire, la seconde moitié de l'ouvrage traite (sans concession) du passage de la version 3 aux versions plus récentes de Turbo-Pascal. Tous les principes présentés dans la première moitié sont universels et facilement adaptables à toute bibliothèque graphique, notamment au module Graph » intégrée désormais à Turbo-Pascal.

L'ouvrage possède un charme indéniable, dû d'une part à la rigueur de son contenu (précisons qu'il est destiné aux élèves de math sup, math spé, et premier cycle universitaire),

Le Pascal Orienté Objet

Depuis la version 5.5 de Turbo-Pascal, on voit fleurir les livres traitant de la programmation orientée objet en Pascal. Les auteurs se sont posés un problème assez surprenant : celui de la sauvegarde directe des objets sur disque.

De mon point de vue, celui d'un programmeur SmallTalk — la référence actuelle pour ce qui est des langages à objets — il s'agit d'un faux problème. Il est indéniable que les structures dynamiques sont toujours difficiles à stocker sur disque. Les objets, comme les pointeurs, sont des structures dynamiques et entrent dans ce cadre, jusque-là tout le monde est d'accord. Parfois, il est indispensable de préserver des données dans des fichiers. L'approche objet serait, au lieu d'essayer de contourner la couche objet pour les sauvegardes, de l'utiliser. Un objet connaît sa structure interne, il est donc capable de placer ses informations sur le disque et de les récupérer plus tard. Il est normal qu'un objet qui puisse être rangé sur disque possède une méthode de stockage et une méthode de restauration.

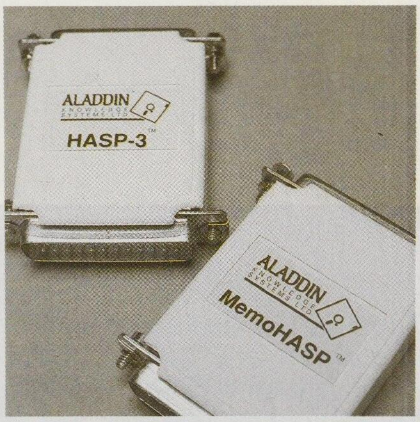
Toutes manipulations jugées utiles seront effectuées par ces méthodes. Par exemple, rien n'empêche (et c'est même facile) de compacter les données avant sauvegarde et de les décompacter à la relecture...

C.G.

BIEN SÛR QUE OUI!

Il y a des dongles (clefs) de protection contre le piratage des logiciels qui

NE SONT PAS PAREILS



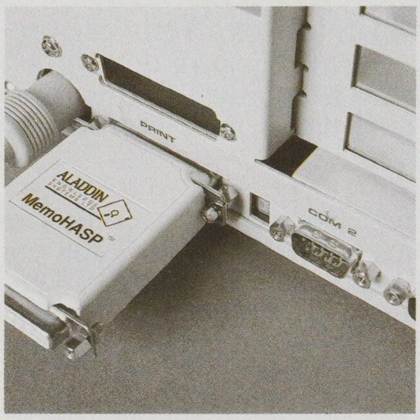
Il y en a qui :
- ont un circuit spécifique avec une électronique plus sophistiquée,
- disposent d'un algorithme plus complexe,
- ont une mémoire plus grande,
- sont plus transparentes,
- sont plus fiables,
- sont virtuellement impossible à déplomber,
- sont munis d'un dispositif "anti-debugging" et de "virus detection",
- sont empilables,

et qui :

COÛTENT MOINS CHER

et surtout

NE CRAIGNENT PAS LA COMPARAISON AVEC LES AUTRES



C'est pourquoi **LOGIDATA** vous offre un **package d'évaluation** de ses dongles HASP à un prix réduit, soit :

HASP-II	**225.– F**	**HASP-3**	**235.– F**
MEMOHASP 1	**295.– F**	**MEMOHASP 4**	**350.– F**

(PORT et TVA compris, contre remboursement plus 30 F)

accompagné d'une disquette d'installation complète.

Les dongles HASP ont été vendues dans 37 pays à plus de 200.000 unités.

NOUVEAU

DESORMAIS, AUSSI DISPONIBLES
SERIES de 5 et 10 dongles AYANT LE MEME CODE.

LOGIDATA / B.P. n° 2 - 74500 PUBLIER / Evian - France
Tél. (33) 50 70 73 75 - Fax (33) 50 75 31 44

Cerclez info pub 32

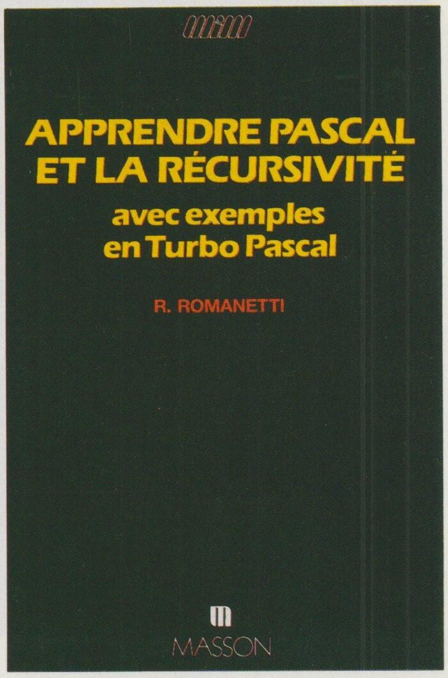


mais surtout à la présentation des limites des procédés informatiques utilisés, ou du langage, et aux nombreuses remarques en notes à la fin des chapitres.

Apprendre Pascal et la récursivité avec exemples en Turbo-Pascal, *de R. Romanetti, aux éditions Masson.* La récursivité est présentée à travers le langage Pascal, sans obscurité ni formalisme excessif (contrairement à Langage Pascal et Logique du Premier Ordre). Ce livre n'est à conseiller qu'à des programmeurs déjà confirmés qui n'hésiteront pas à produire un effort certain. Le résultat peut être fructueux.

Des problèmes classiques illustrent les chapitres, les tours de Hanoi, le parcours du cavalier, les huit reines, etc. En fin d'ouvrage, divers schémas de dérécursivation sont présentés. La récursivité ne devrait plus effrayer aucun programmeur après cette lecture.

Guillaume de Brébisson et Christophe Grosjean



Un Institut réservé aux pascaliens

C'est au 26 de la rue Lamartine, dans le 9e arrondissement, que se trouve l'Institut Pascal, dirigé par John Colibri. Cette petite boutique est exclusivement réservée aux pascaliens de tous horizons, et ne vend de livres que sur le Pascal. Mais toutes les versions du langage (Apple II, Mac, Atari, Quick, Turbo, UCSD, etc.) sont traités, et les livres sont de qualité : de Wirth à Tennenbaum, en passant par Knuth, seuls les bons auteurs (souvent américains) sont représentés.John Colibri n'aime que le Pascal et vit de sa passion.

En 1979, il a été jusqu'à quitter son poste d'ingénieur (il est ancien de Polytechnique) pour pouvoir programmer tranquillement. Depuis, il a écrit près de quinze livres sur le Pascal, dont cinq dédiés au Turbo-Pascal. Il a fondé l'Institut Pascal, où se réunissent tous les mardis d'autres pascaliens fous, a créé la revue Pascalissime (vendue par abonnement), et anime des stages (chers) d'apprentissage du langage.Sa passion est constructrice. Elle lui a permis de devenir un programmeur hors pair, qui a écrit entre autres un compilateur, un assembleur, un Lisp, un Prolog, un désassembleur, le traitement de textes dont il se sert pour écrire ses livres, un générateur d'applications, etc. Tous ces programmes sont en vente à des prix dérisoires à l'Institut Pascal, et sont toujours fournis avec leur code source, malheureusement trop souvent écrit en Turbo version 3.0.

L'intérêt d'aller le rencontrer, c'est qu'il vous conseillera dans vos achats de livres selon vos besoins et votre niveau, et sera toujours prêt à discuter un moment sur la nouvelle version du dernier langage sorti, ou sur un algorithme particulièrement complexe.

G.B

ACTION PC LE MAGAZINE DE LA PRODUCTIVITE AMELIOREE

NE RATEZ PAS NOTRE NUMERO SPECIAL PORTABLES

NOUVEAU

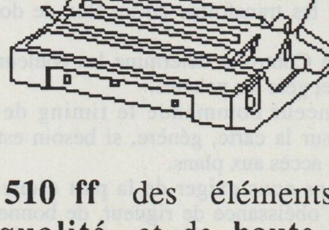
pour **510 ff ht**

TRANSFORMEZ

votre cartouche *ordinaire* d'origine

en une cartouche *professionnelle*

***spéciale* P A O**



Nouveau cylindre d'impression

Nouveau fil corona plaqué OR

Nouveau toner haute résolution

Pour **510 ff** des éléments de **qualité** et de **haute technologie** transforment votre cartouche jetable en un **sous-ensemble d'impression prêt à l'emploi** et recyclable **indéfiniment.**

Les cartouches P A O sont recyclables *indéfiniment* pour 350 ff ht

(1) 30 55 66 84

SERELEC idF

Siége social : 1307 rue de la Boissière- 78370 PLAISIR
Téléphone : (1) 30 55 66 84 - Télécopie : (1) 30 54 03 56

Cerclez info pub 33

Shareware...

Des problèmes de fabrication ont causé des érreurs dans les programmes parus dans le précédent numéro. Nous les publions ci-dessous " débuggés ".

VgaTest

Bien que n'étant pas la norme graphique la plus performante disponible actuellement, le mode VGA s'impose comme standard. Il nous a paru intéressant d'offrir aux lecteurs d'Action PC un outil de test. Plusieurs facteurs expliquent le succès de la norme VGA. La baisse du prix des composants a mis le matériel à la portée du plus grand nombre. Parallèlement, la montée en puissance des machines permet la pleine exploitation de produits comme « Windows » ou « Presentation Manager ». La plupart des logiciels de haut niveau disposent d'un « driver » VGA et on assiste à la multiplication des logiciels spécialement écrits pour fonctionner dans un environnement graphique. Enfin, sur des machines de plus en plus nombreuses, la norme VGA est directement implantée sur la carte mère, ce qui facilite grandement la propagation de la norme. L'émergence de cartes intelligentes, dotées d'un processeur propre, marquera certainement la fin de la norme VGA. Cependant, comme toute norme suffisamment répandue, elle aura une retraite heureuse, et de nombreuses générations de logiciels permettront de l'exploiter.

Rappelons, pour mémoire, quelques spécifications du VGA :
- émulation MDA, CGA, Hercules, EGA,
- mode 12h : 640 x 480 en 16 couleurs parmi 262 144,
- mode 13h : 320 x 200 en 256 couleurs parmi 262 144,
- sortie RVB analogique,
- fonctions booléennes en écriture,
- split screen.

Sur les cartes VGA un peu améliorées, on trouve aussi :
- mode 29h : 800 x 600 en 16 couleurs parmi 262 144,
- zoom matériel (hardware),
- mode texte de 132 colonnes.

Enfin, avec les cartes Super VGA, on dispose de :
- mode 30h : 800 x 600 en 256 couleurs parmi 262 144,
- mode 37h : 1024 x 768 en 16 couleurs parmi 262 144,
- bus interne de 32 bits.

Par rapport aux précédents standards, CGA et EGA, le VGA marque une nette évolution dans la complexité.

Les fonctionnalités, mais aussi les technologies utilisées, diffèrent souvent. Jusqu'à EGA incluse, les cartes graphiques utilisaient une sortie de type TTL : chaque ligne du connecteur pouvait prendre une valeur, 0 ou 1, et le nombre total de couleurs disponibles était de 2n, où n représente le nombre de lignes véhiculant l'information de couleur. Pour l'EGA, 6 lignes du connecteur de 9 broches sont consacrées à la couleur, ce qui donne une palette de 64 couleurs. La sortie analogique de la VGA consacre trois lignes à la couleur, mais il s'agit de lignes analogiques. Chaque composante de couleur est caractérisée par le niveau électrique de sa ligne, et peut prendre, en théorie, une infinité de valeurs. Dans la pratique, seuls 64 niveaux de signaux sont acceptés, ce qui nous donne 643 = 262 144 couleurs disponibles. Pour notre malheur, toutes ces couleurs disponibles ne le sont pas simultanément. En effet, l'information de couleur, pour chaque pixel, doit être stockée en mémoire. Ainsi, pour disposer de 262 144 couleurs simultanées, chaque pixel doit être défini sur 18 bits. Outre que ce format est peu pratique, il faudrait 700 Ko de mémoire pour coder un écran de 640 x 480 pixels. Les fabricants ont donc recours à une table de couleurs, qui associe à un pixel codé sur 4 bits (16 couleurs) ou sur 8 bits (256 couleurs), une couleur réelle, codée sur 18 bits. Prenons le mode 12h : 640 x 480 en 16 couleurs. Les besoins en mémoire sont de 640 x 480 x 4 = 1 228 800 bits, soit environ 150 Ko. La mémoire du PC directement adressable étant toujours de 1 Mo, seuls deux segments de 64 Ko restent disponibles pour l'affichage. L'accès aux données s'effectue donc, comme pour la mémoire paginée, par l'intermédiaire d'une « fenêtre », découpée dans l'un des segments.Deux configurations de la mémoire sont utilisées : le mode plan, et le mode linéaire. En mode plan, chaque pixel correspond à un bit dans chacun des quatre plans. Vu depuis le processeur, chaque plan réside à la même adresse. Les lectures/écritures sont ainsi plus complexes qu'elles ne l'étaient avec la norme CGA. En écriture, il est possible d'accéder simultanément à plusieurs plans par un système de masques. En lecture, par contre, on ne peut en sélectionner qu'un seul. Ce système, un peu pénalisant, a une contrepartie : chaque opération peut concerner jusqu'à huit pixels. De plus, des fonctions comme le « color compare », permettent de contourner, dans certains cas, les limitations en lecture. En mode linéaire, le pixel est codé sur huit bits consécutifs. Un plan unique étant utilisé, chaque lecture ou écriture peut se faire en un seul accès mémoire. La faible résolution du mode 13h provient de cette restriction sur le nombre de plans utilisés : 640 x 200 = 64 Ko. En linéaire, les fonctions booléennes ne sont plus accessibles. Autrement dit, c'est le processeur qui doit réaliser lui-même les OR, AND et autres XOR. Les modes linéaires disponibles sur les VGA améliorées, qui permettent d'afficher jusqu'à 800 x 600 en 256 couleurs réalisent un peu l'amalgame des deux méthodes. Les plans sont chaînés, l'adressage d'un pixel devra alors se faire en deux fois : sélection du plan, puis opération. Les performances s'en trouvent dégradées, l'adressage d'un point prenant le même temps que l'adressage de huit points en mode plan. La carte VGA comporte quelques autres éléments dont nous allons dresser un rapide inventaire :
- le Graphics Controler. Situé entre la mémoire écran et le processeur, il régit les opérations logiques, les comparaisons de couleurs, et les rotations de données.
- le CRT Controler s'occupe principalement des cycles de rafraîchissement du CRT (Cathodic Ray Tube). Il est utilisé pour le zoom matériel, lorsque celui-ci existe.
- le Data Serializer récupère les informations dans les différents plans mémoire et les transforme en un flux de données acceptable par le CRT.
- l'Attribute Controler détermine les couleurs finales du texte et du graphique, et gère les palettes.
- le séquenceur commande le timing de toutes les opérations effectuées sur la carte, génère, si besoin est, des états d'attente, et contrôle les accès aux plans.

Que pouvons-nous exiger de la part d'une carte VGA ? Outre la fidélité et l'obéissance de rigueur, de bonnes performances sont en général les bienvenues. Il faut savoir que, d'un matériel à l'autre, les temps d'affichage peuvent varier du simple au triple, en fonction du temps d'accès à la mémoire, aux registres, de la largeur du bus, etc. Certains logiciels consacrent près de 80 % des ressources machine à gérer l'affichage. On comprend mieux l'intérêt de disposer d'une carte VGA rapide. Le programme VGATEST fournit quatre indices de performance :
- nombre de caractères écrits par seconde, en accès mémoire direct, et par le BIOS,
- nombre de pixels générés par seconde, accès direct et BIOS.

Les valeurs trouvées n'ont réellement un sens qu'en comparaison des

valeurs fournies par d'autres cartes. En effet, des ordres de grandeur tels que 1 000 000 de caractères par seconde ne peuvent être obtenus qu'en expédiant 64 000 fois le même caractère d'un coup, de même pour les pixels. Les résultats obtenus sont loin de refléter les valeurs que l'on peut espérer obtenir lors d'utilisations usuelles. Ecrit en C et en assembleur, VGATEST fournit les outils pour obtenir les renseignements suivants :

- type d'affichage : MDA, CGA, EGA couleur ou mono, VGA couleur ou mono,
- mémoire disponible : jusqu'à 512 Ko de mémoir vive (RAM) peuvent figurer sur la carte.
- largeur du bus : un bus de 16 bits permet d'améliorer les performances de la VGA, avec certaines restrictions. Cette possibilité n'est en effet valide qu'en mode d'adressage linéaire (texte ou graphique 256). Il faut, de plus, que le logiciel utilisé prévoit ses adressages sur 16 bits de données. Il est donc clair qu'une carte disposant d'un bus de 16 bits n'est pas, loin s'en faut, deux fois plus performante qu'une carte limitée à 8 bits.
- la fréquence de rafraîchissement vertical est le nombre de fois où le faisceau d'électrons parcourt totalement l'écran par seconde.
- la fréquence de rafraîchissement horizontal correspond au nombre de fois par seconde où le faisceau d'électrons s'interrompt pour passer de la droite à la gauche de l'écran.
- la taille de la matrice de caractères en mode texte (16 x 8 d'origine).
- le copyright de la carte figure dans les premiers octets de la ROM. Il donne en général le nom du constructeur, la date de fabrication et la version du BIOS VGA.

Quelques conclusions peuvent être tirées de nos essais de VGATEST :

- les circuits VGA intégrés sur la carte mère sont en général beaucoup plus rapides que les cartes additionnelles,
- le BIOS supplémentaire du type FASTBIOS, livré avec la carte, est deux à trois fois plus performant que le BIOS qui figure sur la carte elle-même,
- la pleine utilisation d'un bus de 16 bits améliore de 60 à 70 % la rapidité de l'affichage linéaire, Le tableau ci-dessous offre un aperçu de quelques valeurs que nous avons rencontrées. On notera les écarts étonnants qui existent d'un matériel à l'autre.

MATERIEL	CARACTERES ACCES DIRECT	CARACTERES BIOS	PIXELS ACCES DIRECT	PIXELS BIOS
QUADRAM GTI 16	629 000	151 000	652 900	2450
QUADRAM GTI 16 FASTBIOS	629 000	498 000	652 900	7690
COMPAQ CARTE	2 194 000	306 000	1 876 900	14 300
MERE 286e VGA PARADISE	741 000	84 000	1 054 200	4340

VGATEST.C

```
*/
#include <dos.h>
#include <conio.h>

/*
Nombres d'itérations pour chaque test : diminuer ces
valeurs augmente la rapidité du test au détriment de
sa précision
*/
#define ITER_VERT_RETRACE 300
#define ITER_HORIZ_RETRACE 32000
#define ITER_AFF_DIR_GRAPH 30
#define ITER_AFF_DIR_TXT 40
#define ITER_AFF_BIOS_GRAPH 30
#define ITER_AFF_BIOS_TXT 10

/* des retour */
#define OK 0
#define NOT_VGA 1

/* Distinguer le vrai du faux */
#define TRUE  1
#define FALSE 0

/* Adresses mémoire */
#define OFS_REALTIME_LOW  0x046C
#define OFS_REALTIME_HIGH 0x046E
```

```
*****************************************************
*              VGATEST                  1.00        *
*****************************************************
*                                                   *
*                                                   *
* = FONCTION   :        Banc test pour cartes VGA   *
*                                                   *
* = DATE          :        04-90                    *
*                                                   *
* = AUTEUR   : Jean-Michel SPINER                   *
*              pour ACTION PC                       *
*                                                   *
*****************************************************
```

```
#define BIOS_SEG      0x0000
#define BIOS_CARTE_SEG          0xC000
#define COPYRIGHT_OFS           0x0000
#define BIOS_BPC      0x0485

/* Largeur car en mode 3 */
#define L_STANDARD 8

/* Types possibles d'affichage */
#define  MDA          1
#define  CGA          2
#define  ENHANCED 3
#define  EGACOUL  4
#define  EGAMONO  5
#define  VGAMONO  7

#define  VGACOUL  8

/* Modes */
#define MODE_TXT8025_16 0x03
#define MODE_640480_16 0x12
#define MODE_800600_256 0x30

/* Couleurs d'affichage */
#define BLANC_SUR_BLEU 0x1F
#define NOIR_SUR_GRIS 0x70
#define VERT_SUR_BLEU 0x1A
#define CLIGNO_BLANC_SUR_BLEU 0x9F

/* Adresses des registres VGA */
#define INPUT_STATUS_REGISTER_COUL 0x03DA
#define INPUT_STATUS_REGISTER_MONO 0x03BA
unsigned int input_status_register;

/* Adresses des segments textes */
#define SEG_TXT_COUL 0xB800
#define SEG_TXT_MONO 0xB000
unsigned int adr_txt;

/* Pointeurs sur le Bios */
unsigned far *ptr_realtime_low;
unsigned far *ptr_realtime_high;
unsigned far *ptr_bpc;
unsigned char far *ptr_copyright;

/* Ptr sur le status info */
#define OFS_EQUIP 0x0487
#define SEG_EQUIP 0x0000
unsigned char far *ptr_equip;

/* Variables de configuration */
unsigned char display_type;
unsigned mem_vga;
unsigned vertical_retrace;
unsigned horizontal_retrace;
unsigned nb_bits;
unsigned car_l,car_h;
float aff_dir_graph;
float aff_dir_txt;
float aff_bios_graph;
float aff_bios_txt;

/*              PROTOTYPES             */
void Init_vars(unsigned char display_type);
void Write_Copyright (void);
int Get_Display_Type(void);
int Get_Mem_Vga(void);
unsigned Get_Vertical_Retrace(void);
unsigned Get_Horizontal_Retrace(void);
void Set_Screen_Mode (unsigned char mode,unsigned char cls);
unsigned Get_Screen_Mode (void);
void Put_Pixel_Bios (unsigned x,unsigned y,unsigned coul);
unsigned Get_Pixel_Bios (unsigned x,unsigned y);
void Chrono (int *h,int *m,int *s,int *c);
float Get_Aff_Dir_Txt (void);
float Get_Aff_Bios_Txt (void);
float Get_Aff_Dir_Graph (void);
float Get_Aff_Bios_Graph (void);
void Get_Car_lh (unsigned *l,unsigned *h);
unsigned Get_Nb_Bits (void);
char *Format(unsigned long val,char *str10,unsigned long arrondi);

/****************************/
/*                    MAIN                    */
/***************************/
int main (void)
{
   char str10[10];
   char texte[80];
   unsigned long l;
   float fl;

   /* Initialisations */
   FP_SEG (ptr_equip) = SEG_EQUIP;
   FP_OFF (ptr_equip) = OFS_EQUIP;
   display_type = Get_Display_Type();
   Set_Screen_Mode (MODE_640480_16,TRUE);
   if (display_type<7) {
```

```
      printf ("\ntype %d : Adaptateur VGA non trouvé\n\n",display_type);
      exit (NOT_VGA);
   }
   Init_vars(display_type);
   /*       mode 640 480    */
   Set_Screen_Mode (MODE_640480_16,TRUE);
   aff_bios_graph = Get_Aff_Bios_Graph ();
   Set_Screen_Mode (MODE_640480_16,TRUE);
   aff_dir_graph = Get_Aff_Dir_Graph ();
   /*       mode texte             */
   Set_Screen_Mode (MODE_TXT8025_16,TRUE);
   aff_bios_txt = Get_Aff_Bios_Txt ();
   aff_dir_txt = Get_Aff_Dir_Txt ();
   Set_Screen_Mode (MODE_TXT8025_16,TRUE);
   nb_bits = Get_Nb_Bits ();
   mem_vga = Get_Mem_Vga ();
   /*          MISE EN PLACE DE L'AFFICHAGE           */
   Set_Screen_Mode (MODE_TXT8025_16,TRUE);
   printf ("\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n");
   print (" **************************************-¬ ",BLANC_SUR_BLEU,0,0);
   print (" |                                                                        | ",BLANC_SUR_BLEU,0,1);
   print ("     VGATEST  V1.00                       Par JM SPINER pour ACTION PC ",NOIR_SUR_GRIS,2,1);
   print (" |*****************|********************| ",BLANC_SUR_BLEU,0,2);
   print (" |      CARACTERISTIQUES                   PERFORMANCES                  | ",BLANC_SUR_BLEU,0,3);
   print (" |*****************+********************| ",BLANC_SUR_BLEU,0,4);
   print (" |                                   |                                    | ",BLANC_SUR_BLEU,0,5);
   print (" | Type     :                        | Vitesse d'affichage en mode texte    | ",BLANC_SUR_BLEU,0,6);
   print (" |                                   | ****************        | ",BLANC_SUR_BLEU,0,7);
   print (" | Mémoire  :                        | Par accès mémoire :                  | ",BLANC_SUR_BLEU,0,8);
   print (" |                                   | Par le bios       :                  | ",BLANC_SUR_BLEU,0,9);
   print (" | Bus      :                        |                                      | ",BLANC_SUR_BLEU,0,10);
   print (" |                                   |                                      | ",BLANC_SUR_BLEU,0,11);
   print (" | Balayage vertical   :             | Vitesse d'affichage en mode graphique  | ",BLANC_SUR_BLEU,0,12);
   print (" |                                   | ******************  | ",BLANC_SUR_BLEU,0,13);
   print (" | Balayage horizontal :             | Par accès mémoire :                  | ",BLANC_SUR_BLEU,0,14);
   print (" |                                   | Par le bios       :                  | ",BLANC_SUR_BLEU,0,15);
   print (" | Matrice caractère   :             |                                      | ",BLANC_SUR_BLEU,0,16);
   print (" |                                   |                                      | ",BLANC_SUR_BLEU,0,17);
   print (" |*****************|********************| ",BLANC_SUR_BLEU,0,18);
   print (" | Copyright :                                                            | ",BLANC_SUR_BLEU,0,19);
   print (" |                                                                        | ",BLANC_SUR_BLEU,0,20);
   print (" |                                                                        | ",BLANC_SUR_BLEU,0,21);
   print (" |************************************** ",BLANC_SUR_BLEU,0,22);
   /* Type de display */
   switch (display_type) {
      case MDA     : strcpy (texte,"MDA"); break;
      case CGA     : strcpy (texte,"CGA"); break;
      case EGACOUL : strcpy (texte,"EGA couleur"); break;
      case EGAMONO : strcpy (texte,"EGA monochrome"); break;
      case VGAMONO : strcpy (texte,"VGA monochrome"); break;
      case VGACOUL : strcpy (texte,"VGA couleur"); break;
   }
   print (texte,VERT_SUR_BLEU,15,6);
   /* Mémoire */
   sprintf (texte,"%d Ko",mem_vga);
   print (texte,VERT_SUR_BLEU,15,8);
   /* Bus */
   sprintf (texte,"%d Bits",nb_bits);
   print (texte,VERT_SUR_BLEU,15,10);
   /* Matrice de car */
   Get_Car_lh (&car_l,&car_h);
   sprintf (texte,"%dx%d",car_h,car_l);
   print (texte,VERT_SUR_BLEU,31,16);

   /* Affichage texte par accès mémoire */
   strcpy (texte,format((long)aff_dir_txt,str10,1000L));
   strcat (texte," car./s");
   print (texte,VERT_SUR_BLEU,59,8);
   /* Affichage texte par bios */
   strcpy (texte,format((long)aff_bios_txt,str10,1000L));
   strcat (texte," car./s");
   print (texte,VERT_SUR_BLEU,59,9);

   /* Affichage pixels par accès mémoire */
   strcpy (texte,format((long)aff_dir_graph,str10,1000L));
   strcat (texte," pxl./s");
   print (texte,VERT_SUR_BLEU,59,14);

   /* Affichage pixels par bios */
   strcpy (texte,format((long)aff_bios_graph,str10,100L));
   strcat (texte," pxl./s");
   print (texte,VERT_SUR_BLEU,61,15);

   /* Copyright */
   Write_Copyright ();
   /* Vertical retrace */
   print ("> WAIT <",CLIGNO_BLANC_SUR_BLEU,27,12);
   vertical_retrace = Get_Vertical_Retrace();
   sprintf (texte,"%5d Hz",vertical_retrace);
   print (texte,VERT_SUR_BLEU,27,12);
   /* Horizontal retrace */
   print ("> WAIT <",CLIGNO_BLANC_SUR_BLEU,27,14);
   horizontal_retrace = Get_Horizontal_Retrace ();
   strcpy (texte,format((long)horizontal_retrace,str10,100L));
   strcat (texte," Hz");
   print (texte,VERT_SUR_BLEU,26,14);
   exit (OK);
}
```

```
/*******************************/
/*                    INIT_VARS                          */
/*******************************/
void Init_vars(unsigned char display_type)
{
   FP_OFF (ptr_realtime_low) = OFS_REALTIME_LOW;
   FP_OFF (ptr_realtime_high) = OFS_REALTIME_HIGH;
   FP_SEG (ptr_realtime_low) = BIOS_SEG;
   FP_SEG (ptr_realtime_high) = BIOS_SEG;
   FP_SEG (ptr_bpc) = BIOS_SEG;
   FP_OFF (ptr_bpc) = BIOS_BPC;
   FP_SEG (ptr_copyright) = BIOS_CARTE_SEG;
   FP_OFF (ptr_copyright) = COPYRIGHT_OFS;
   if (display_type==VGAMONO) {
      input_status_register = INPUT_STATUS_REGISTER_MONO;
      adr_txt = SEG_TXT_MONO;
   }
   else {
      input_status_register = INPUT_STATUS_REGISTER_COUL;
      adr_txt = SEG_TXT_COUL;
   }
}
/*******************************/
/*                 WRITE_COPYRIGHT                    */
/*******************************/
void Write_Copyright (void)
{
   register r1;
   char ch;
   char texte[2];
   int lgn,col;

   col = 15; lgn = 19;
   if ((*(ptr_copyright++)!=0x55)||(*(ptr_copyright++)!=0xAA)) {
      print (“Non Trouvé”,VERT_SUR_BLEU,col,lgn);
   }
   else {
      for (r1=0;r1<160;r1++) {
         ch = *(ptr_copyright++);
         if ((ch>31)&&(ch<126)) {
            texte[0] = ch; texte[1] = ‘\0’;
            print (texte,VERT_SUR_BLEU,col,lgn);
            col++;
            if (((ch==32)&&(col>67))||(col>75)) {
               col = 3;
               lgn ++;
            }
         }
      }
   }
}
/*******************************/
/*               GET_DISPLAY_TYPE                     */
/*******************************/
int Get_Display_Type(void)
{
   union REGS regs;

   regs.h.ah = 0x1A;
   regs.h.al = 0x00;
   int86 (0x10,&regs,&regs);
   if (regs.h.al==0x01) {
      return (MDA);
   }
   if (regs.h.al==0x02) {
      return (CGA);
   }
   if (regs.h.al==0x1A) {
      /* VGA */
      if ( (*(ptr_equip) && 0x02) == 0x02 ) {
         return (VGAMONO);
      }
      else {
         return (VGACOUL);
      }
   }
   regs.h.ah = 0x12;
   regs.h.bl = 0x10;
   int86 (0x10,&regs,&regs);
   if (regs.h.bh==0x01) {
      /* MONO */
      return (EGAMONO);
   }
   if ((regs.h.cl==0x09)||(regs.h.cl==0x03)) {
      /* ENHANCED */
      return (ENHANCED);
   }
   /* EGACOUL */
   return (EGACOUL);
}
/*******************************/
/*                GET_MEM_VGA                            */
/*******************************/
int Get_Mem_Vga(void)
{
   register r1;
   union REGS regs;
```

```
   regs.h.ah = 0x12;
   regs.h.bl = 0x10;
   int86 (0x10,&regs,&regs);
   rl = regs.h.bl;
   /* 512 K ? */
   if (rl>=3)
   {
      Set_Screen_Mode(MODE_800600_256,TRUE);
      Put_Pixel_Bios (750,550,200);
      if (Get_Pixel_Bios(750,550) == 200) {
         rl = 7;
      }
   }
   return ((rl+1)*64);
}
/********************************/
/*                       GET_VERTICAL_RETRACE                 */
/********************************/
unsigned Get_Vertical_Retrace(void)
{
   register i1;
   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;

   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<ITER_VERT_RETRACE;i1++) {
      Wait_Vert_Retrace ();
   }
   chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps = fin - debut;
   return_value = ITER_VERT_RETRACE / tps;
   if ((return_value-(int)return_value)>0.5) return_value++;
   return ( (int)(return_value) );
}
/********************************/
/*                       GET_HORIZONTAL_RETRACE               */
unsigned Get_Horizontal_Retrace(void)
{
   unsigned i1;

   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;

   chrono (&h1,&m1,&s1,&c1);

   for (i1=0;i1<ITER_HORIZ_RETRACE;i1++) {
      Wait_Horiz_Retrace ();
   }

   chrono (&h2,&m2,&s2,&c2);

   debut = (float)(h1*36000.0 + m1*60.0 + s1 + c1/100.0);
   fin = (float)(h2*36000.0 + m2*60.0 + s2 + c2/100.0);
   tps = fin - debut;
   return_value = ITER_HORIZ_RETRACE / tps;
   if ((return_value-(int)return_value)>0.5) return_value++;
   return ( (int)(return_value) );
}
/*                          SET_SCREEN_MODE                     */
void Set_Screen_Mode (unsigned char mode,unsigned char cls)
{
   union REGS regs;

   if (cls==FALSE) {
      mode |= 0x80;
   }
   regs.h.ah = 0x00;
   regs.h.al = mode;
   int86 (0x10,&regs,&regs);
}
/*                          GET_SCREEN_MODE                     */
unsigned Get_Screen_Mode (void)
{
   union REGS regs;

   regs.h.ah = 0x0F;
   int86 (0x10,&regs,&regs);
   regs.h.ah = 0x00;
   return (regs.x.ax);
}
/*                          PUT_PIXEL_BIOS                      */
void Put_Pixel_Bios (unsigned x,unsigned y,unsigned coul)
{
   union REGS regs;

   regs.x.ax = coul;
   regs.h.ah = 0x0C;
   regs.x.bx = 0x0000;
   regs.x.cx = x;
   regs.x.dx = y;
   int86 (0x10,&regs,&regs);
}
```

```
/*                      GET_PIXEL_BIOS                              */
unsigned Get_Pixel_Bios (unsigned x,unsigned y)
{
   union REGS regs; regs.h.ah = 0x0D;
   regs.x.bx = 0x0000;
   regs.x.cx = x;
   regs.x.dx = y;
   int86 (0x10,&regs,&regs);
   regs.h.ah = 0x0000;
   return (regs.x.ax);
}
/*                      CHRONO                                      */
void Chrono (int *h,int *m,int *s,int *c)
{
   union REGS regs;

   regs.h.ah = 0x2C;
   int86 (0x21,&regs,&regs);
   *h = regs.h.ch;
   *m = regs.h.cl;
   *s = regs.h.dh;
   *c = regs.h.dl;
}
/*                      GET_AFF_DIR_GRAPH                           */
float Get_Aff_Dir_Graph (void)
{
   unsigned register i1;
   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;

   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<ITER_AFF_DIR_GRAPH;i1++) {
      Write_Pxl_Dir (i1);
   }
   chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps = fin - debut;
   return_value = (ITER_AFF_DIR_GRAPH * 32767.0) / tps;
   return (return_value);
}

/*                      GET_AFF_BIOS_GRAPH                          */
float Get_Aff_Bios_Graph (void)
{
   unsigned register i1;
   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;

   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<ITER_AFF_BIOS_GRAPH;i1++) {
      Write_Pxl_Bios (i1);
   }
   chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps = fin - debut;
   return_value = ITER_AFF_BIOS_GRAPH * 640.0 / tps;
   return (return_value);
}
/*                      GET_AFF_DIR_TXT                             */
float Get_Aff_Dir_Txt (void)
{
   unsigned register i1;
   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;
   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<ITER_AFF_DIR_TXT;i1++) {
      Write_Char_Dir_W (i1);
   }
   chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps = fin - debut;
   return_value = (ITER_AFF_DIR_TXT * 32767.0) / tps;
   return (return_value);
}
/*                      GET_AFF_BIOS_TXT                            */
float Get_Aff_Bios_Txt (void)
{
   unsigned register i1;
   float debut,fin,tps;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;
   float return_value;

   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<ITER_AFF_BIOS_TXT;i1++) {
      Write_Char_Bios (i1);
   }
   chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
```

```
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps = fin - debut;
   return_value = (ITER_AFF_BIOS_TXT * 32767.0) / tps;
   return (return_value);
}
/*                    GET_CAR_LH                        */
void Get_Car_lh (unsigned *l,unsigned *h)
{
   *h = *(ptr_bpc);
   *l = L_STANDARD;
}
/*                   GET_NB_BITS                        */
unsigned Get_Nb_Bits (void)
{
   unsigned register i1;
   float debut,fin,tps1,tps2;
   int h1,m1,s1,c1;
   int h2,m2,s2,c2;

   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<30;i1++) {
      Write_Char_Dir_B (i1);
   }

 chrono (&h2,&m2,&s2,&c2);
   debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps1 = fin - debut;
   chrono (&h1,&m1,&s1,&c1);
   for (i1=0;i1<30;i1++) {
      Write_Char_Dir_W (i1);
   }

 debut = (float)h1*36000.0 + (float)m1*60.0 + (float)s1 + (float)c1/100.0;
   fin = (float)h2*36000.0 + (float)m2*60.0 + (float)s2 + (float)c2/100.0;
   tps2 = fin - debut;
   if ((tps2-tps1)<(tps1/3.0)) {
      return (16);
   }
   else {
      return (8);
   }
}
/*                   FORMAT                        */

char *Format(unsigned long val,char *str10,unsigned long arrondi)
{
   char milion,mil;
   char *strtmp[10];
   long l1;

   strset (str10,'\0');
   l1 = val / arrondi;
   l1 *= arrondi;

   milion = FALSE;

  mil = FALSE;
   if (l1>=1000000L) {
      milion = TRUE;
      sprintf (strtmp,"%lu ",l1/1000000L);
      strcat (str10,strtmp);
      l1 = l1 % 1000000L;
      if (l1<1000L) strcat (str10,"000 ");
   }
   if (l1>=1000L) {
      mil = TRUE;
      if (milion == TRUE) {
         if (l1<100000L) strcat (str10,"0");
         if (l1<10000L) strcat (str10,"0");
      }
      sprintf (strtmp,"%lu ",l1/1000L);
      strcat (str10,strtmp);
      l1 = l1 % 1000L;
   }

if ( (milion==TRUE) || (mil==TRUE) ) {
      if (l1<100L) strcat (str10,"0");
      if (l1<10L) strcat (str10,"0");
   }
   sprintf (strtmp,"%lu",l1);
   strcat (str10,strtmp);
   return (str10);
}

~Ç@
   print (texte,VERT_SUR_BLEU,15,6);
   /* Mémoire */
   sprintf (texte,"%d Ko",mem_vga);
   print
```

VGATOOLS.ASM 1.00

```
AUTEUR   : Jean-Michel SPINER
pour ACTION PC
;public _Print

public   _Wait_Horiz_Retrace
public   _Wait_Vert_Retrace
public   _Write_Char_Dir_B
public   _Write_Char_Dir_W
public   _Write_Char_Bios

public   _Write_Pxl_Dir
public   _Write_Pxl_Bios
                 .286
_DATA    segment word public 'DATA'
```

```
extrn    _input_status_register:word
extrn    _adr_txt:word
_DATA    ends

DGROUP   group _DATA

TEXT     segment byte public 'CODE'
         assume  ds:DGROUP, cs:_TEXT
;                     PRINT
_Print proc near
sframe        struc
              dw        ?
              dw        ?
txt           dw        ?
attrib        dw        ?
col           dw        ?
lgn           dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              push      di
              push      si
              mov       si,frame.txt
              mov       es,_adr_txt
              mov       di,frame.lgn
              mov       ax,di
              shl       di,7
              shl       ax,5
              add       di,ax
              mov       ax,frame.col
              shl       ax,1
              add       di,ax
              mov       ah,byte ptr
frame.attrib
              cld
lascar:
              cmp       byte ptr [si],0
              je        finitos
              lodsb
              stosw
              jmp       lascar
finitos:
              pop       si
              pop       di
              mov       sp,bp
              pop       bp
              ret

_Print endp

_Wait_Horiz_Retrace proc near
        push    es
        mov
dx,_input_status_register
        in      al,dx
        test    al,01h
        jz      H_Ready
H_Not_Ready:
        in      al,dx
        test    al,01h
        jnz     H_Not_ready
H_Ready:
        in      al,dx
        test    al,01h
        jz      H_Ready
        pop     es
        ret
_Wait_Horiz_Retrace endp

        WAIT_VERT_RETRACE
_Wait_Vert_Retrace  proc near
        push    es
        mov
dx,_input_status_register
        in      al,dx
        jmp     $+2
        test    al,08h
        jz      V_Ready
V_Not_Ready:
        in      al,dx
        jmp     $+2
        test    al,08h
        jnz     V_Not_ready
V_Ready:
        in      al,dx
        jmp     $+2
        test    al,08h
        jz      V_Ready
        pop     es
        ret
_Wait_Vert_Retrace    endp

        WRITE_CHAR_DIR_B
_Write_Char_Dir_B     proc near
sframe        struc
              dw        ?
              dw        ?
charb         dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              ; empilages
              push      di
              push      es
              mov       di,0000h
              ; segment dans es
              mov       es,_adr_txt
              ; Çcrire
              mov       ax,frame.charb
              mov       ah,0Fh
              mov       cx,07FFFh
              cld
              rep       stosb
              pop       es
              pop       di
              mov       sp,bp
              pop       bp
              ret
_Write_Char_Dir_B     endp
        WRITE_CHAR_DIR_W
_Write_Char_Dir_W     proc near
sframe        struc
              dw        ?
              dw        ?
charw         dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              ; empilages
              push      di
              push      es
              mov       di,0000h
              ; segment dans es
              mov       es,_adr_txt
              ; Çcrire
              mov       ax,frame.charw
              mov       ah,0Fh
              mov       cx,07FFFh
              cld
              rep       stosw
              pop       es
              pop       di
              mov       sp,bp
pop      bp
              ret
_Write_Char_Dir_W     endp
        WRITE_CHAR_BIOS
_Write_Char_Bios     proc near
sframe        struc
              dw        ?
              dw        ?
bchar         dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              mov       ax,frame.bchar
              mov       ah,0Ah
              mov       bh,00h
              mov       cx,07FFFh
              int       10h

              mov       sp,bp
              pop       bp
              ret
_Write_Char_Bios        endp

                  WRITE_PXL_DIR
_Write_Pxl_Dir          proc     Near
sframe        struc
              dw        ?
              dw        ?
val           dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              ; Empilages
              push      di
              push      si
              push      es
              mov       di,0000h
              ; faire pointer ds sur
segment video
              mov       ax,0A000h
              mov       es,ax
              ; Autorizer les plans
              mov       dx,03C4h
              mov       al,02h
              out       dx,al
              inc       dx
              mov       al,byte ptr
frame.val
              and       al,07h
              out       dx,al
              mov       ax,0FFFFh
              mov       cx,07FFFh
              cld
              rep       stosb
              pop       es
              pop       si
              pop       di
              mov       sp,bp
              pop       bp
              ret
_Write_Pxl_Dir          endp

                  WRITE_PXL_BIOS
; _Write_Pxl_Bios         proc     Near
sframe        struc
              dw        ?
              dw        ?
coul          dw        ?
sframe        ends
frame         equ       [bp]
              push      bp
              mov       bp,sp
              push      si
              push      di
              mov       cx,0000h
              mov       dx,240
              mov       bh,0000h
boucle:
              mov       ax,frame.coul
              mov       ah,0Ch
              int       10h
              inc       cx
              cmp       cx,640
              jnz       boucle
              pop       di
              pop       si
              mov       sp, bp
              pop       bp
              ret
_Write_Pxl_Bios         endp
```

KOPY

Les quatre commandes du DOS les plus utilisées sont sûrement « DIR », « CD », « DEL » et « COPY ». Après vous avoir présenté des versions améliorées de CD et DEL, respectivement CDV et KILL dans Action PC n°3, et une version déguisée et générale de DIR, le programme WHEREIS dans Action PC n°4, voici une version améliorée de la commande COPY. Cette nouvelle mouture permet de copier toute une arborescence de fichiers dans une autre. Le programme commence comme d'habitude par analyser la ligne de commande. Celle-ci est de la forme « Kopy [répertoire]<nom> <répertoire> [/<options>] ». Le premier paramètre représente la source. Des « wildcards » peuvent être utilisés pour désigner un ensemble de fichiers. Si aucun répertoire n'est précisé, ce sont les fichiers du répertoire courant qui seront copiés. Il est possible de n'indiquer qu'un répertoire, les fichiers de ce répertoire seront alors copiés (la syntaxe « répertoire*.* » est équivalente à la notation « répertoire »). Le deuxième paramètre est le répertoire de destination. Ce paramètre peut être omis, les fichiers seront copiés dans le répertoire courant. Les options sont au nombre de quatre et ne doivent pas être séparées par des espaces après le signe « / ». La première est l'option « a » qui permet d'indiquer une copie à partir de tous les disques durs. Par exemple, si sur chacun des disques durs, le répertoire TEXTE contient des documents, il est possible de dupliquer tous les documents dans un nouveau répertoire (NEWDIR) avec la commande : « KOPY TEXTE*.DOC NEWDIR /a ». La deuxième option « m » indique un déplacement des fichiers plutôt qu'une copie. Le fichier source est effacé après sa copie. Par défaut, tous les fichiers contenus dans la sous-arborescence du répertoire source sont aussi copiés en reproduisant la sous-arborescence. La troisième option est l'option « s » et cette option indique de ne pas copier la sous-arborescence (contrairement à « xcopy »). La dernière option « p » entraîne une mise à plat des fichiers lors de la copie. C'est-à-dire que les fichiers de la sous-arborescence du répertoire source seront copiés directement dans le répertoire destination et non dans un sous-répertoire. Par exemple, la commande : « KOPY *.C SOURCE /AP », copie tous les fichiers de tous les disques durs se terminant par l'extension .C dans le répertoire SOURCE. Alors que la commande : « KOPY *.C SOURCE /A », recopie tous les fichiers de tous les disques durs d'extension .C en reproduisant l'arborescence dans laquelle les fichiers se trouvaient, sous le répertoire SOURCE. Après l'analyse de la ligne de commande, les chemins des répertoires source et destination sont compactés pour supprimer les répertoires « .. » et obtenir ainsi en référence absolue (et non relative) des répertoires. Cette phase est nécessaire pour remédier au principal problème posé par le programme. A savoir de ne pas effectuer une copie infinie des fichiers. Il ne faut pas copier dans un sous-répertoire du répertoire destination le répertoire destination lui-même et copier dans un sous-répertoire de ce sous-répertoire, ce sous-répertoire, etc. (Vous me suivez bien ?). La copie bouclant ainsi jusqu'à saturation du disque dur. Pour éviter une telle situation, le programme arrête l'exploration des sous-répertoires dès que les répertoires source et destination sont identiques. Cette comparaison s'effectuant sur des chaînes de caractères, il faut donc bien que les chemins soit des références absolues. Une fois les chemins source et destination obtenus, le programme s'organise principalement autour des fonctions « chercher_fichier » et « copy ». La première cherche les fichiers à copier alors que la deuxième les copie. On observe dans la fonction « chercher_fichier », l'utilisation des options « s » et « p ». la première option empêche la recherche dans la sous-arborescence et la deuxième interdit la modification du répertoire de destination. Ainsi, on s'aperçoit qu'il est inutile de préciser l'option « p » si l'option « s » figure déjà dans la ligne de commande.

La fonction « copy » réalise simplement la copie grâce aux fonctions « open », « read », « write » et « close », la constante TAILLE_BUFF déterminant la taille du buffer à utiliser. Si l'option « m » est indiquée, le fichier source est supprimé à la fin de la copie. Les fonctions « fopen », « fread », « fwrite » et « fclose » n'ont pas été utilisées car elles différencient mal une fin de fichier et une erreur. Si une erreur se produit durant la copie, le fichier destination est effacé pour ne pas laisser de fichier incomplet sur le disque. La fonction « verif_dest » est appelée pour vérifier si le répertoire destination existe. Si ce répertoire n'existe pas, il est créé, ainsi que tous les répertoires pères nécessaires. Lorsque le travail du programme est terminé, la fonction « fin » est appelée. Elle affiche le nombre de répertoires créés et le nombre de fichiers copiés. L'utilisation des fonctions « findfirst » et « findnext » entraîne l'inclusion du fichier DOS.H. Pour ne pas rencontrer de problème lors de la compilation, il est indispensable de positionner les options suivantes à « off » : dans options/compiler/source, « ANSI keywords only » et dans options/link, « case-sensitive link ». La compilation peut être opérée avec le modèle de mémoire « tiny », et le programme ainsi obtenu transformé en « .com » grâce à exe2bin.

Vincent Pouilley

```
_
TEXT      ends
end
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <dir.h>
#include <string.h>
#include <sys\stat.h>
#include <fcntl.h>

/* constantes */
#define TAILLE_NOM          13
#define TAILLE_PATH                   128
#define TAILLE_BUFF                   32000
#define VRAI                          1
#define FAUX                          0

/* variables */
char chercher[TAILLE_NOM]; /* nom du fichier cherché */
char ref_dest[TAILLE_PATH];
char opta, optm, opts, optp;
long nbrep = 0;
long nbfic = 0;

void usage(commande)
char * commande;
{
        char * paux, * ptemp;
        char nom_prg[80];

        paux = strrchr(commande,'\\');
        if (paux == NULL)
                paux = commande;
        else
                paux++;
```

```
        ptemp = nom_prg;
        while((*ptemp++ = *paux++) != '.');
        *(ptemp - 1) = '\0';
        printf("%s V1.11 : Copyright 1990 Vincent POUILLEY\n",nom_prg);
        puts("Cette commande permet de copier des fichiers");
        puts("dans une arborescence\n");
        printf("usage : %s [répertoire]<nom> <répertoire> [/<options>]\n",nom_prg);
        puts("    nom  : nom du fichier");
        puts("options  :");
        puts("      a  : copie à partir de tous les disques durs");
        puts("      m  : déplace les fichiers au lieu de les copier");
        puts("      s  : ne copie pas les sous-répertoires");
        puts("      p  : copie les fichiers mais pas l'arborescence");
}

void fin(err)
int err;
{
        printf("     %d répertoire(s) créé(s)\n",nbrep);
        if (optm)
                printf("     %d fichier(s) déplacé(s)\n",nbfic);
        else
                printf("     %d fichier(s) copié(s)\n",nbfic);
        exit(-1);
} /* fin */

void verif_dest(dest)
char dest[];
{
        char * ptemp, * ptemp2, sauve;
        struct stat etat;

        ptemp = dest + (strlen(dest) - 1);
        if (strlen(dest) > 3)
                *ptemp = '\0';
        else
                if (access(dest,0) < 0)
                        {
                                printf("La destination %s n'est pas valide\n",dest);
                                fin(-1);
                        }
                else
                        return;
        if (stat(dest,&etat) == 0)
                {
                *ptemp = '\\';
if ((etat.st_mode & S_IFDIR) == S_IFDIR)
                        return;
                printf("La destination %s est un fichier\n",dest);
                fin(-1);
                }
        ptemp2 = ptemp;
        while((ptemp2 > dest + 2) && (*(--ptemp2) != '\\'));
        sauve = *(++ptemp2);
        *ptemp2 = '\0';
        verif_dest(dest);
        *ptemp2 = sauve;
        if (mkdir(dest) != 0)
                {
                        printf("Création du répertoire %s impossible\n",dest);
                        fin(-1);
                }
        printf("Création du répertoire %s\n",dest);
        nbrep++;
        *ptemp = '\\';
} /* verif_dest */

void copy(src,dest,nom)
char * src, * dest , * nom;
{
        int ficent, ficsort;
        char buff[TAILLE_BUFF], * psrc, * pdest;
        int nr, nw;

        psrc  = src  + strlen(src);
        pdest = dest + strlen(dest);
        strcpy(psrc ,nom);
        strcpy(pdest,nom);
        if (optm)
                printf("Transfert de %s vers %s\n",src,dest);
        else
                printf("Copie de %s vers %s\n",src,dest);
        ficent  = open(src,O_RDONLY | O_BINARY);
        if (ficent < 0)
                {
                        printf("erreur de lecture du fichier %s\n",src);
                        fin(-1);
                }
        ficsort = open(dest,O_WRONLY | O_CREAT | O_TRUNC | O_BINARY,S_IWRITE);
        if (ficsort < 0)
                {
                        printf("erreur d'écriture du fichier %s\n",dest);
```

```
		close(ficent);
			fin(-1);
		}
	do
		{
			nr = read(ficent,buff,TAILLE_BUFF);
			if (nr < 0)
				{
					printf("erreur de lecture du fichier %s\n",src);
					close(ficent);
					close(ficsort);
					unlink(dest);
					fin(-1);
				}
			nw = write(ficsort,buff,nr);
			if (nw != nr)
				{
					printf("erreur d'écriture du fichier %s\n",dest);
					close(ficent);
					close(ficsort);
					unlink(dest);
					fin(-1);
				}
		}
	while(nr == TAILLE_BUFF);
	close(ficent);
	close(ficsort);
	if (optm)
		unlink(src);
	*psrc  = '\0';
	*pdest = '\0';
	nbfic++;
} /* copy */

void chercher_fichier(src, dest)
char * src, * dest;
{
	struct ffblk ffblk;
	char * tsrc, * tdest;
	int err;

	if (strcmp(src,ref_dest) == 0)
		return;
	tsrc = src + strlen(src);
	if (!opts)
		{
			strcpy(tsrc,"*.*");
			err = findfirst(src,&ffblk,FA_DIREC);
			*tsrc = '\0';
			while(!err)
				{
					if ((strcmp(ffblk.ff_name,".") != 0) &&
						(strcmp(ffblk.ff_name,"..") != 0) &&
						((ffblk.ff_attrib & FA_DIREC) == FA_DIREC))
						{
							strcpy(tsrc,ffblk.ff_name);
							strcat(tsrc,"\\");
							if (!optp)
								{
									tdest = dest + strlen(dest);
									strcpy(tdest,ffblk.ff_name);
									strcat(tdest,"\\");
								}
							chercher_fichier(src, dest);
							*tsrc = '\0';
							if (!optp)
								*tdest = '\0';
						}
					err = findnext(&ffblk);
				} /* while */
		} /* if */
	strcpy(tsrc,chercher);
	err = findfirst(src,&ffblk,FA_ARCH | FA_RDONLY | FA_HIDDEN |
		FA_SYSTEM);
	*tsrc = '\0';
	if (!err)
		verif_dest(dest);
	while(!err)
		{
			copy(src,dest,ffblk.ff_name);
			err = findnext(&ffblk);
		}
} /* chercher fichier */

int compact_path(path)
char * path;
{
	char * paux, * ptemp;

	paux = path + strlen(path);
	if (*(paux - 1) != '\\')
		strcpy(paux,"\\");
	paux = path;
	while (*paux != '\0')
		{
			if ((*paux == '.') && (*(paux + 1) == '.'))
				{
					ptemp = paux - 2;
					while((*(-ptemp) != '\\') && (ptemp > path));
```

```
                                        if (ptemp <= path)
                                                return(-1);
                                        strcpy(ptemp + 1,paux + 3);
                                        paux = ptemp;
                                }
                        paux++;
                }
        return(0);
} /* compact path */

main(argc,argv)
int argc;
char * argv[];
{
        char src[TAILLE_PATH], drive[3], dir[66], name[9], ext[5];
        int drivedest, j, k, drive_courant, drivesrc, nbdrive;
        char * paux, * paux2, * paux3, dest[TAILLE_PATH], ptemp[66];
        struct stat etat;

        drive_courant = getdisk();
        nbdrive = setdisk(drive_courant);
        strcpy(src ,"C:\\");
        strcpy(dest,"C:\\");
        opta = optm = opts = optp = FAUX;
        for(k = 1; k < argc; k++)
                {
                        paux = argv[k];
                        for(j = 0; paux[j] != '\0'; j++)
                                paux[j] = toupper(paux[j]);
                }
        switch (argc)
                {
                case 2 :
                                paux = argv[1];
                                paux2 = strchr(argv[1],'/');
                                if (paux2 != NULL)
                                        {
                                                *paux2 = '\0';
                                                while(*(++ paux2) != '\0')
                                                        switch (*paux2)
                                                                {
                                                                case 'A' :
                                                                        opta = VRAI;
                                                                        break;
                                                                case 'M' :
                                                                        optm = VRAI;
                                                                        break;
                                                                case 'S' :
                                                                        opts = VRAI;
                                                                        break;
                                                                case 'P' :
                                                                        optp = VRAI;
                                                                case '/' :
                                                                        break;
                                                                default :
                                                                        usage(argv[0]);
                                                                        exit(-1);
                                                                }
                                        }
                                paux2 = paux + strlen(paux);
                                break;
                case 3 :
                                paux2 = strchr(argv[2],'/');
                                if (paux2 != NULL)
                                        {
                                                *paux2 = '\0';
                                                while(*(++ paux2) != '\0')
                                                        switch (*paux2)
                                                                {
                                                                case 'A' :
                                                                        opta = VRAI;
                                                                        break;
                                                                case 'M' :
                                                                        optm = VRAI;
                                                                        break;
                                                                case 'S' :
                                                                        opts = VRAI;
                                                                        break;
                                                                case 'P' :
                                                                        optp = VRAI;
                                                                case '/' :
                                                                        break;
                                                                default :
                                                                        usage(argv[0]);
                                                                        exit(-1);
                                                                }
                                        }
                                paux2 = argv[2];
                                paux = argv[1];
                                break;
                case 4 :
                                paux  = argv[1];
                                paux2 = argv[2];
                                if (argv[3][0] != '/')
                                        {
                                                usage(argv[0]);
                                                exit(-1);
                                        }
                                paux3 = argv[3];
```

```
                        while(*(++ paux3) != '\0')
                                switch (*paux3)
                                        {
                                                case 'A' :
                                                        opta = VRAI;
                                                        break;
                                                case 'M' :
                                                        optm = VRAI;
                                                        break;
                                                case 'S' :
                                                        opts = VRAI;
                                                        break;
                                                case 'P' :
                                                        optp = VRAI;
                                                case '/' :
                                                        break;
                                                default :
                                                        usage(argv[0]);
                                                        exit(-1);
                                        }
                        break;
                default :
                        usage(argv[0]);
                        exit(-1);
        } /* switch */
/* analyse de la source */
        j = fnsplit(paux,drive,dir,name,ext);
        if ((j & DRIVE) == DRIVE)
                drivesrc = toupper(drive[0]) - 'A';
        else
                drivesrc = drive_courant;
        src[0] = drivesrc + 'A';
        if ((j & DIRECTORY) == DIRECTORY)
                if (dir[0] == '\\')
                        {
                                src[2] = '\0';
                                strcat(src,dir);
                        }
                else
                        {
                                k = getcurdir(drivesrc+1,ptemp);
                                if (k)
                                        {
                                                puts("erreur de lecture");
                                                fin(-1);
                                        }
                                strcat(src,ptemp);
                                if (ptemp[0] != '\0')
                                        strcat(src,"\\");
                                if (strcmp(dir,".") != 0)
                                        strcat(src,dir);
                        }
        else
                {
                        k = getcurdir(drivesrc+1,ptemp);
                        if (k)
                                {
                                        puts("erreur de lecture");
                                        fin(-1);
                                }
                        strcat(src,ptemp);
                        if (ptemp[0] != '\0')
                                strcat(src,"\\");
                }
        if ((j & FILENAME) == FILENAME)
                {
                        paux = src + strlen(src);
                        strcpy(paux,name);
                        strcat(paux,ext);
                        if ((stat(src,&etat) == 0) &&
                           ((etat.st_mode & S_IFDIR) == S_IFDIR) &&
                           (strchr(name,'*') == NULL) && (strchr(ext,'*') == NULL))
                                {
                                        strcpy(chercher,"*.*");
                                        strcat(paux,"\\");
                                }
                        else
                                {
                                        *paux = '\0';
                                        strcpy(chercher,name);
                                        strcat(chercher,ext);
                                }
}
        else
                strcpy(chercher,"*.*");
        if (compact_path(src) < 0)
                {
                        puts("Répertoire source inexistant");
                        exit(-1)
                }
/* analyse de la destination */
        j = fnsplit(paux2,drive,dir,name,ext);
        if ((j & DRIVE) == DRIVE)
                drivedest = toupper(drive[0]) - 'A';
        else
                drivedest = drive_courant;
        dest[0] = drivedest + 'A';
```

```
if ((j & DIRECTORY) == DIRECTORY)
        if (dir[0] == '\\')
                {
                        dest[2] = '\0';
                        strcat(dest,dir);
                }
        else
                {
                        k = getcurdir(drivedest+1,ptemp);
                        if (k)
                                {
                                        puts("erreur de lecture");
                                        fin(-1);
                                }
                        strcat(dest,ptemp);
                        if (ptemp[0] != '\0')
                                strcat(dest,"\\");
                        if (strcmp(dir,".") != 0)
                                strcat(dest,dir);
                }
    else
        {
                k = getcurdir(drivedest+1,ptemp);
                if (k)
                        {
                                puts("erreur de lecture");
                                fin(-1);
                        }
                strcat(dest,ptemp);
                if (ptemp[0] != '\0')
                        strcat(dest,"\\");
        }
    if ((j & FILENAME) == FILENAME)
        {
                if (dest[strlen(dest) - 1] != '\\')
                        strcat(dest,"\\");
                strcat(dest,name);
                strcat(dest,ext);
                strcat(dest,"\\");
        }
    if (compact_path(dest) < 0)
        {
                puts("Répertoire destination impossible");
                exit(-1);
        }
    strcpy(ref_dest,dest);
    if (opta)
        {
                if (drivesrc < 2)
                        chercher_fichier(src, dest);
                for(drivesrc = 2; drivesrc < nbdrive; drivesrc++)
                        {
                                src[0] = drivesrc + 'A';
                                chercher_fichier(src, dest);
                        }
        }
    else
        chercher_fichier(src, dest);
    fin(0);
} /* main */
```

Recherche de chaîne Grep

La commande « grep » est bien connue des programmeurs sous Unix. Elle recherche une chaîne quelconque de caractères dans des fichiers. Sa fonction peut être, par exemple, lors du développement de gros programmes séparés en modules, de retrouver tous les appels et la déclaration d'une fonction, ou de tracer l'emploi d'une variable. Son intérêt n'est pas limité aux programmeurs. « Grep » peut aussi, entre autres, déterminer la liste des textes traitant d'un même sujet. Pour les utilisateurs du traitement de textes Word, on retrouve cette fonction à partir de la version 4.
Pour utiliser le programme, la syntaxe est :
« GREP <chaîne> <fichier> /options ».

<chaîne> représente la chaîne de caractères. Pour rechercher une chaîne contenant des espaces, il faut mettre ce paramètre entre guillemets. Dans cette version du grep, la chaîne est recherchée telle quelle, c'est-à-dire qu'il n'est pas possible de mettre de « wildcards » pour une recherche plus générale.
<fichier> indique le fichier dans lequel la recherche doit s'effectuer. Le nom du fichier peut être précédé d'un chemin et il peut contenir les wildcards « * » et « ? ». Seuls les fichiers du répertoire indiqué sont pris en compte.
Enfin, les options sont au nombre de six. On retrouve l'habituelle option « a » qui indique un parcours de tous les disques durs, l'option « p » qui attend la pression d'une touche après chaque page de 25 lignes, et l'option « s » qui indique une recherche dans les fichiers contenus dans la sous-arborescence. Les trois dernières options sont plus spécifiques au programme. L'option « i » indique une recherche sans tenir compte des majuscules et des minuscules. L'option « n » provoquera l'affichage du numéro de ligne dans le fichier où se trouve la chaîne. L'option « m » empêche l'affichage de la ligne du fichier à l'écran. La notion de ligne est ici prise au sens de paragraphe, c'est-à-dire jusqu'au retour chariot. A la fin de la recherche, le programme affiche le nombre d'occurrences de la chaîne, le nombre de fichiers où elle a été trouvée, et le nombre de fichiers examinés. Le nombre d'occurrences n'est pas tout à fait exact car le programme indique plus précisément, en fait, le nombre de lignes. Le programme possède la même structure que « Kopy » pour la partie recherche de fichiers. La fonction « main » analyse la ligne de commande, puis appelle la fonction « cherche_fichier ». Celle-ci réalise le parcours du répertoire pour trouver les fichiers concernés et, si l'option « s » est utilisée, réalise le parcours de l'arborescence de manière récursive. Pour chaque fichier, elle fait appel à la fonction « cherche_chaine » qui lit le fichier ligne à ligne. Elle utilise également la fonction « trouver » et gère l'affichage. La fonction « trouver » explore la ligne à la recherche de la chaîne. L'algorithme utilisé est celui de Boyer Moore. Il serait possible, pour accélérer la recherche, d'implémenter l'algorithme de Moris et Pratt. Cependant, pour une utilisation sur des programmes, les lignes n'ont en général pas plus de quatre-vingts caractères et, en conséquence, le temps de recherche devient négligeable devant le temps de lecture du fichier. Le programme a été réalisé en Turbo C et utilise la fonction « fnsplit ». Si vous possédez un autre compilateur, il faut employer le fichier « fnsplit.h » contenant une définition de cette fonction. Il faut ajouter, dans la section des includes, la ligne : « #include "fnsplit.h" ». La compilation peut être réalisée avec le modèle de mémoire « tiny » et le programme transformé en « .com » grâce à l'utilitaire « exe2bin » du DOS.

Vincent Pouilley

```
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <dir.h>
#include <string.h>
#include <sys\stat.h>
#include <conio.h>

/* constantes */
#define TAILLE_NOM              13
#define TAILLE_PATH             128
#define TAILLE_LIGNE            20000
#define VRAI                            1
#define FAUX                            0

/* variables */
char fichier[TAILLE_NOM];
char * chercher; /* chaine cherché */
char opta, optm, optn, opts, optp, opti;
int  nbligne, nbocc, nbfic, nbfictot;
int color_direct, color_inverse;

void usage(commande)
char * commande;
{
          char * paux, * ptemp;
          char nom_prg[80];

          paux = strrchr(commande,'\\');
          if (paux == NULL)
                    paux = commande;
          else
                    paux++;
          ptemp = nom_prg;
          while((*ptemp++ = *paux++) != '.');
          *(ptemp - 1) = '\0';

printf("%s V1.1 : Copyright 1990 Vincent

POUILLEY\n",nom_prg);
          puts("Cette commande permet de rechercher une chaîne de caractères");
          puts("dans un ou plusieurs fichiers\n");
          printf("usage : %s <chaine> [<répertoire>]<nom> [/<options>]\n",nom_prg);
          puts(" chaine : chaîne de caractères à rechercher");
          puts("    nom : nom du fichier");
          puts("options :");
          puts("    a   : recherche sur tous les disques durs");
          puts("    n   : affiche les numéros de ligne dans le fichier");
          puts("    s   : recherche dans les sous-répertoires");
          puts("    m   : pas d'affichage de la ligne du fichier");
          puts("    i   : ne distingue pas les majuscules des minuscules");
          puts("    p   : pause après chaque page");
}

fin()
{
          if (optp && (nbligne >= 22))
                    {
                              cprintf("appuyer sur une touche");
                              getch();
                              cprintf("\r");
                    }
          cprintf("     '%s' trouvé %d fois dans %d fichier(s) parmi %d fichier(s)\r\n"
                    ,chercher,nbocc,nbfic,nbfictot);
          exit(0);
}
int trouver(ligne,ls,lc)
char * ligne;
int ls, lc;
{
          int i, j;

          i = j = 0;
          if (opti)
                    while (i+lc <= ls)
                              {
                                        while ((j < lc) && (toupper(ligne[i+j]) == toupper(chercher[j])))
                                                  j++;
                                        if (j < lc)
                                                  {
                                                            i++;
                                                            j = 0;
                                                  }
                                        else
                                                  return(i);
                              }
          else
                    while (i+lc <= ls)
                              {
                                        while ((j < lc) && (ligne[i+j] == chercher[j]))
                                                  j++;
                                        if (j < lc)
                                                  {
                                                            i++;
                                                            j = 0;
                                                  }
                                        else
                                                  return(i);
                              }
          return(-1);
}

cherche_chaine(rep,nom)
char * rep, * nom;
{
          FILE * fic;
          long noligne;
          int  ls, lc, nbcar, pos, lg, lf, ap, av;
          char * temp, * paux, ligne[TAILLE_LIGNE], numero[10], dedans;

          dedans = FAUX;
          noligne = 0;
          lc = strlen(chercher);
          temp = rep + strlen(rep);
```

```
	strcpy(temp,nom);
	fic = fopen(rep,"rt");
	while (!feof(fic))
		{
			if (fgets(ligne,TAILLE_LIGNE,fic) == NULL)
				break;
			if ((ls = strlen(ligne)) != TAILLE_LIGNE -1)
				noligne++;
			if ((pos = trouver(ligne,ls,lc)) != -1)
				{
					dedans = VRAI;
					nbocc++;
					if (optp)
						{
							nbligne++;
							if (nbligne >= 24)
								{
									nbligne = 0;
									cprintf("appuyer sur une touche");
									nbcar = getch();
									cprintf("\r");
									if (nbcar == 3)
										fin();
								}
						}
					if (optm)
						{
							if (opts)
								puts(rep);
							else
								puts(nom);
							break;
						}
					paux = ligne -1;
					while (*(++paux) != '\0')
						if (*paux == '\t')
							*paux = ' ';
						else
							if (*paux == '\n')
								{
									*paux = '\0';
									break;
								}
					if (opts)
						{
							cprintf("%s:",rep);
							nbcar = strlen(rep) + 1;
						}
					else
						{
							cprintf("%s:",nom);
							nbcar = strlen(nom) + 1;
						}
					if (optn)
						{
							cprintf("%d:",noligne);
							sprintf(numero,"%d",noligne);
							nbcar += strlen(numero) + 1;
						}
					ligne[pos] = '\0';
					if (nbcar + ls >= 80)
						{
							lg = (79 - nbcar -lc)>>1;
							lf = ls - pos - lc;
							if (lf < lg)
								{
	ap = lf;
									av = (lg<<1) - lf;
								}
							else
								if (pos < lg)
									{
										av = pos;
										ap = 79 - pos - lc - nbcar;
									}
								else
									ap = av = lg;
							ligne[pos + lc + ap] = '\0';
						}
					else
						av = pos;
					cprintf("%s",ligne + (pos - av));
					textattr(color_inverse);
					cprintf("%s",chercher);
					textattr(color_direct);
					cprintf("%s\r\n",ligne + (pos + lc));
				}
		}
	fclose(fic);
	*temp = '\0';
	if (dedans)
		nbfic++;
}

chercher_fichier(rep)
char * rep;
{
	struct ffblk ffblk;
	char * temp;
	int err;

	temp = rep + strlen(rep);
	strcpy(temp,fichier);
	err = findfirst(rep,&ffblk,FA_ARCH | FA_RDONLY | FA_HIDDEN |
		FA_SYSTEM);
	*temp = '\0';
	while(!err)
		{
			cherche_chaine(rep,ffblk.ff_name);
			nbfictot++;
			err = findnext(&ffblk);
		}
	if (opts)
		{
			strcpy(temp,"*.*");
```

```
            err = findfirst(rep,&ffblk,FA_DIREC);
            *temp = ‘\0’;
            while(!err)
                {
                    if ((strcmp(ffblk.ff_name,”.”) != 0) &&
                            (strcmp(ffblk.ff_name,”..”) != 0) &&
                            ((ffblk.ff_attrib & FA_DIREC) == FA_DIREC))
                        {
                            strcpy(temp,ffblk.ff_name);
                            strcat(temp,”\\”);
                            chercher_fichier(rep);
                            *temp = ‘\0’;
                        }
                    err = findnext(&ffblk);
                }
        }
} /* chercher_fichier */

int compact_path(path)
char * path;
{
    char * paux, * ptemp;

    paux = path + strlen(path);
    if (*(paux - 1) != ‘\\’)
        strcpy(paux,”\\”);
    paux = path;
    while (*paux != ‘\0’)
        {
            if ((*paux == ‘.’) && (*(paux + 1) == ‘.’))
                {
                    ptemp = paux - 2;
                    while((*(--ptemp) != ‘\\’) && (ptemp > path));
                    if (ptemp <= path)
                        return(-1);
                    strcpy(ptemp + 1,paux + 3);
                    paux = ptemp;
                }
            paux++;
        }
    return(0);
} /* compact path */

main(argc,argv)
int argc;
char * argv[];
{
    int i, j, k, nbdrive;
    char rep[TAILLE_PATH], drive[3], dir[66], name[9], ext[5];
    char * paux, * paux2, ptemp[66];
    struct stat etat;
    struct text_info info_texte;

    opta = optm = optn = opts = opti = optp = FAUX;
    nbligne = nbocc = nbfic = nbfictot = 0;
    i = getdisk();
    nbdrive = setdisk(i);
    strcpy(rep,”C:\\”);
/* mémorise les couleurs des caractères */
    gettextinfo(&info_texte);
    color_direct = info_texte.attribute;
/* inversion vidéo, le mot représentant l’attribut est complémenté
   sauf le bit de poids fort qui représente le clignotement */
    color_inverse = ((color_direct >> 4) | (color_direct << 4)) & 0x7F;
    for(k = 2; k < argc; k++)
        {
            paux = argv[k];
            for(j = 0; paux[j] != ‘\0’; j++)
                paux[j] = toupper(paux[j]);
        }
    chercher = argv[1];
    switch (argc)
        {
            case 2 :
                paux = NULL;
                paux2 = strchr(argv[1],’/’);
                if (paux2 != NULL)
                    {
                        *paux2 = ‘\0’;
                        while(*(++ paux2) != ‘\0’)
                            switch (*paux2)
                                {
                                    case ‘A’ :
                                    case ‘a’ :
                                        opta = VRAI;
                                        break;
                                    case ‘M’ :
                                    case ‘m’ :
                                        optm = VRAI;
                                        break;
                                    case ‘N’ :
                                    case ‘n’ :
                                        optn = VRAI;
                                        break;
                                    case ‘S’ :
                                    case ‘s’ :
                                        opts = VRAI;
                                        break;
                                    case ‘I’ :
                                    case ‘i’ :
                                        opti = VRAI;
    break;
                                    case ‘P’ :
                                    case ‘p’ :
                                        optp = VRAI;
                                    case ‘/’ :
                                        break;
                                    default :
                                        usage(argv[0]);
                                        exit(-1);
                                }
                    }
                break;
            case 3 :
                paux  = argv[2];
```

```
            paux2 = strchr(argv[2],'/');
            if (paux2 != NULL)
                {
                    *paux2 = '\0';
                    while(*(++ paux2) != '\0')
                        switch (*paux2)
                            {
                                case 'A' :
                                    opta = VRAI;
                                    break;
                                case 'M' :
                                    optm = VRAI;
                                    break;
                                case 'N' :
                                    optn = VRAI;
                                    break;
                                case 'S' :
                                    opts = VRAI;
                                    break;
                                case 'I' :
                                    opti = VRAI;
                                    break;
                                case 'P' :
                                    optp = VRAI;
                                case '/' :
                                    break;
                                default :
                                    usage(argv[0]);
                                    exit(-1);
                            }
                }
            break;
        case 4 :
            paux  = argv[2];
            if (argv[3][0] != '/')
                {
                    usage(argv[0]);
                    exit(-1);
                }
            paux2 = argv[3];
            while(*(++ paux2) != '\0')
                switch (*paux2)
                    {
                        case 'A' :
                            opta = VRAI;
                            break;
                        case 'M' :
                            optm = VRAI;
                            break;
                        case 'N' :
                            optn = VRAI;
                            break;
                        case 'S' :
                            opts = VRAI;
                            break;
                        case 'I' :
                            opti = VRAI;
                            break;
                        case 'P' :
                            optp = VRAI;
                        case '/' :
                            break;
                        default :
                            usage(argv[0]);
                            exit(-1);

                    }
            break;
        default :
            usage(argv[0]);
            exit(-1);
    } /* switch */
if (paux == NULL)
    j = 0;
else
    j = fnsplit(paux,drive,dir,name,ext);
if ((j & DRIVE) == DRIVE)
    i = toupper(drive[0]) - 'A';
rep[0] = i + 'A';
if ((j & DIRECTORY) == DIRECTORY)
    if (dir[0] == '\\')
        {
            rep[2] = '\0';
            strcat(rep,dir);
        }
    else
        {
            k = getcurdir(i+1,ptemp);
            if (k)
                {
                    puts("erreur de lecture");
                exit(-1);
                }
            strcat(rep,ptemp);
            if (ptemp[0] != '\0')
                strcat(rep,"\\");
            if (strcmp(dir,".") != 0)
                strcat(rep,dir);
        }
else
    {
        k = getcurdir(i+1,ptemp);
        if (k)
            {
                puts("erreur de lecture");
                exit(-1);
            }
        strcat(rep,ptemp);
        if (ptemp[0] != '\0')
            strcat(rep,"\\");
    }
if ((j & FILENAME) == FILENAME)
    {
        paux2 = rep + strlen(rep);
        strcpy(paux2,name);
        strcat(paux2,ext);
```

```
                if (stat(rep,&etat) == 0)
                    {
                        if ((etat.st_mode & S_IFDIR) == S_IFDIR)
                            {
                                strcpy(fichier,"*.*");
                                strcat(paux2,"\\");
                            }
                        else
                            {
                                *paux2 = '\0';
                                strcpy(fichier,name);
                                strcat(fichier,ext);
                            }
                    }
                else
                    if ((j & WILDCARDS) == WILDCARDS)
                        {
                            *paux2 = '\0';
                            strcpy(fichier,name);
                            strcat(fichier,ext);
                        }
                    else
                        {
                            printf("Le fichier %s n'existe pas\n",rep);
                            exit(-1);
                        }
            }
    else
            strcpy(fichier,"*.*");
    if (compact_path(rep) < 0)
            {
                puts("Répertoire inéxistant");
                exit(-1);
            }
    if (opta)
            {
                if (i < 2)
                    chercher_fichier(rep);
                for(i = 2; i < nbdrive; i++)
                    {
                        rep[0] = i + 'A';
                        chercher_fichier(rep);
                    }
            }
    else
            chercher_fichier(rep);
    fin();
}
                ret |= DIRECTORY;
                if (dir != NULL)
                    {
                        for(;(ptemp != paux)&&(i < MAXDIR-2);ptemp++,i++)
                            dir[i] = *ptemp;
                        dir[i++] = *ptemp;
                        dir[i] = '\0';
                    }
                while(ptemp++ != paux);
            }
    if (*ptemp == '.')
            if (ptemp[1] == '.')
                {
                    ret |= DIRECTORY;
                    if (dir  != NULL)
                        {
                            dir[i] = dir[i+1] = '.';
                            dir [i+2] = '\0';
                        }
                    ptemp += 2;
                }
            else
                if (ptemp[1] == '\0')
                    {
                        ret |= DIRECTORY;
                        if (dir != NULL)
                            {
                                dir[i] = '.';
                                dir[i+1] = '\0';
                            }
                        ptemp++;
                    }
    paux = strpbrk(ptemp,"*?");
    if (paux != NULL)
            ret |= WILDCARDS;
    if ((*ptemp != '.') && (*ptemp != '\0'))
            {
                ret |= FILENAME;
                if (name != NULL)
                    {
                        for(i = 0;(*ptemp != '.') && (*ptemp != '\0') &&
                                    (i < MAXFILE - 1);i++,ptemp++)
                            name[i] = *ptemp;
                        name[i] = '\0';
                    }
                while((*ptemp != '.') && (*ptemp != '\0'))
                    ptemp++;
            }
    else
            if (name != NULL)
                name[0] = '\0';
    if (*ptemp == '.')
            {
                ret |= EXTENSION;
                if (ext != NULL)
                    {
                        for(i = 0;(*ptemp != '\0')&&(i < MAXEXT - 1);i++,ptemp++)
                            ext[i] = *ptemp;
                        ext[i] = '\0';
                    }
            }
    else
            if (ext != NULL)
                ext[0] = '\0';
    return(ret);
} /* fnsplit */
```

Conquests of Camelot

Ce Sierra vous siéra

Une nouvelle fois Sierra on-line est à l'honneur dans cette rubrique. Quoi de plus normal quand on observe - que dis-je ? quand on admire leur production.

Editeur : Sierra-on-line
Créateur : Sierra
Support : 4X3" 1/2 ; 10X5" 1/4
Compatibilité : PC, XT, AT
Sortie : en vente
Config.512 KO

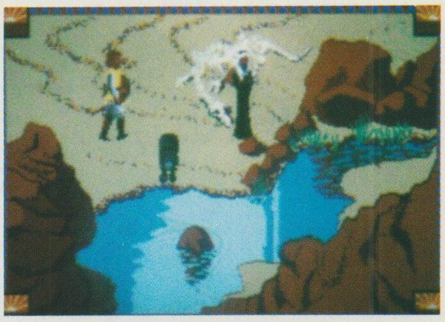
Graphismes : C,E,H,T
Type : aventure
Prix : env. 450 F
Protégé : non
Joueur(s) : 1
Langue : anglais

Note globale : 17/20
Animation : 15
Graphisme : 16
Intérêt : 18
Son : 9

La technologie progresse, les ordinateurs se perfectionnent, et Sierra s'adapte à cette évolution. La généralisation de jeux offrant des graphismes en mode VGA 256 couleurs, et l'adoption du CD-ROM comme support, témoigne de la vitalité de l'éditeur américain.
« Conquests of Camelot » ne bénéficie pas encore du VGA, mais l'EGA lui sied parfaitement, l'augmentation sensible de la taille des personnages contribuant à l'équilibre des graphismes. Et, comme à l'accoutumé, le logiciel exploite les capacités de chaque système : de nombreuses interfaces sonores sont disponibles (MT32, Ad Lib, Game blaster, IBM Music Cards, etc.), tous les modes graphiques sont gérés (VGA excepté), et une souris peut se substituer au clavier pour un grand nombre de commandes.
« Conquests of Camelot » fonctionne sur n'importe quel PC disposant d'au moins 512 Ko. Saluons aussi l'effort de Sierra dans le domaine artistique. Ainsi, après que Bob Siebenberg (Supertramp) et William Goldstein se soient vu confier la composition des musiques de « Space Quest III » et « King's Quest IV », Christy Marx, plus connu pour la série animée de télévision « Jem », a réalisé les graphismes de « Conquest of Camelot ». Ce recours à des talents reconnus explique en grande partie l'excellence des jeux Sierra.



La vision du graal



méfiéz vous du guide et ... de l'eau

le roi, la reine, l'amant

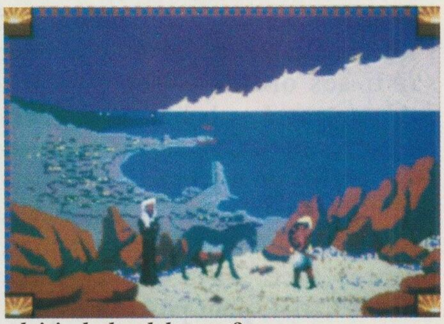
Le scénario de « Conquests of Camelot » s'inspire de l'histoire du royaume de Camelot, dirigé par le roi Arthur (que vous incarnez avec un talent sans égal, bien entendu). L'action se déroule en l'an 932. Le royaume que vous avez bâti prospère de jour en jour, et votre bonheur n'est troublé que par le profond chagrin qui vous habite, votre reine (Guenièvre) étant prise d'amour pour votre plus valeureux chevalier : Lancelot.Or, un jour, le drame s'abat sur votre beau royaume. La sécheresse s'installe, et la famine se fait jour. Au bord du gouffre, il ne subsiste qu'une chance de vaincre la malédiction qui s'acharne sur votre royaume : trouver le Graal.
Seul le Graal, objet divin et fabuleux qui inspire les rêves et les légendes, peut aider à guérir le coeur d'un roi (vous) et apporter un souffle de vie sur une terre mourante (la vôtre). Vous envoyez donc vos trois plus valeureux chevaliers (Lancelot, Galehaut, Gauvain) à sa recherche. Le temps passant, et les émissaires ne revenant pas, vous décidez alors



retour sur la terre ferme



reprenez des forces avant le désert !



plaisir du bord de mer ?

de partir à votre tour en quête de l'objet merveilleux. Avec l'aide de Merlin l'enchanteur qui vous accompagnera par l'esprit.

« Conquests of Camelot » constitue un jeu d'aventure assez difficile, qui se déroule en trois actes majeurs. Le premier consiste à préparer la quête du Graal. Si cette étape ne pose guère de problème, il convient toutefois de ne rien oublier, car une fois la grille du château franchie, vous ne pouvez y retourner qu'en possession du Graal. Durant cette première partie, vous apprendrez ce qui est survenu à vos preux chevaliers et une vision du Graal vous apparaîtra.

La deuxième partie de l'aventure vous convie à délivrer les trois valeureux (incapables ?) chevaliers, retenus prisonniers. Vous devez pour cela affronter bien des adversaires, comme le Chevalier noir, défié au cours d'une joute — ce passage est d'ailleurs assez bien réalisé.

Comme dans « Hero's Quest », l'aventure est émaillée de petits jeux d'arcade, de faible qualité mais distrayants tout de même. Ceux qui trouveront la partie arcade trop ardue ne manqueront pas d'actionner l'option (utile) qui permet de définir le degré de difficulté de l'épreuve selon trois niveaux : facile, normal, difficile.
D'autres pièges guettent le vaillant roi : la sorcière, puis la Demoiselle de glace (Ice Maiden), représentation dépourvue d'amour de la Dame du lac (!)... ainsi que des énigmes (connaissez-vous, par exemple, le langage des fleurs ?).
Cette deuxième étape est aussi la plus difficile. Les énigmes sont nombreuses (et posées en anglais), les obstacles difficiles à surmonter. En particuliers la traversée du Lac gelé, qui demande beaucoup de patience et de nombreuses sauvegardes.

Les combats assez nombreux témoignent du chaos de ces temps troublés...Une fois ces contrariétés éludées, il ne reste que l'ultime acte : la traversée du désert, à pied et dans la bonne humeur. Durant cette épreuve, six « juges » testent vos qualités morales et physiques. Et ce n'est qu'à l'issue de ce jugement que vous atteindrez votre but : le Graal. Mais la traversée de l'Erg se révèle périlleuse, et trouver un bon guide est fondamental. Un « tuyau », ne lui faites pas confiance et surtout ne buvez pas ce que votre mule refuse !

la quête du Graal

La difficulté de « Conquests of Camelot » lui confère une « durée de vie » importante, sans pour autant que le jeu devienne fastidieux. Graphiquement, il se distingue des précédentes productions par le recours à des personnages plus grands et plus soignés. Contrairement à « Colonel's Bequest », « Conquests of Camelot » ne se déroule pas en temps réel. Par contre, et c'est une bonne chose, un effort a été fourni dans le scénario pour empêcher que le joueur ne soit trop souvent bloqué, ayant omis une action précédemment — il faut alors, en effet, remonter assez loin dans l'action pour corriger l'oubli. C'est donc un retour à une technique plus ancienne des jeux de Sierra.Un problème subsiste, se faire comprendre par l'ordinateur. Il arrive encore qu'un ordre soit mal interprété. « Conquests of Camelot » reprend l'interface étrennée par Sierra depuis toujours, c'est-à-dire que le joueur dispose

les combats sont rudes et nombreux



la jonte



le puit et son trésor



d'une fenêtre de commandes, par l'intermédiaire de laquelle il peut saisir ses ordres — en anglais, évidemment. Les habituels raccourcis subsistent (Ctrl-I pour l'inventaire, Ctrl-A pour formuler les questions, F5 pour la sauvegarde, etc.), auxquels se greffent quelques autres, propres à ce jeu : il s'agit des touches Ctrl-T, pour poser une question à Merlin, et Ctrl-O, pour ouvrir sa bourse (eh oui ! une nouvelle fois, après « Hero's Quest », l'argent joue ici un rôle important).Ce qui surprend le plus, une fois le jeu chargé, c'est de constater la suppression de la petite bande en haut de l'écran, qui indiquait le score et les menus, offrant par la même occasion une image plus grande et carrée. Alors qu'en matière d'audiovisuel la haute résolution arrive avec les écrans rectangulaires, chez Sierra on semble vouloir accompagner la haute résolution (VGA) d'images carrées...Toutefois, la disparition de cette bande n'est que partielle, et on peut l'afficher en appuyant sur la touche « échappement ». Comme dans « Hero's Quest », il n'y a plus un score mais trois, qui varient selon votre comportement durant le jeu. Cependant, il ne s'agit pas de faire évoluer un personnage à la manière d'un jeu de rôle (points de vie, etc.) mais plutôt d'une décomposition du score en trois catégories, afin de juger les valeurs de votre âme (soul), de votre habileté (skill), et de votre sagesse (wisdom) — ce qui est nécessaire pour savoir si vous méritez ou non de posséder le Graal. Les personnages affichés étant plus gros, l'image plus grande, et les détails plus nombreux, il est préférable de posséder un PC muni d'un processeur performant, cadencé au moins à 12 MHz (et encore !). Dans le cas contraire, l'animation risque de pêcher par une lenteur (très) irritante, et ce en dépit de l'option permettant d'accélérer ou de ralentir le personnage. Il est aussi recommandé d'installer le jeu sur disque dur, en raison de sa taille, 3 Mo de fichiers (sans compter les sauvegardes !).



le palace de la glace

de 7 à 77 ans

« Conquests of Camelot » reprend une technique plus classique, dans sa réalisation, que « Colonel's Bequest » ou « Hero's Quest », ce qui devrait plaire aux habitués des jeux Sierra. Dans la lignée des Sierra plus anciens, « Conquests of Camelot » bénéficie d'une longue expérience de la part des programmeurs, ce qui en fait une réalisation de très bonne qualité, malgré quelques légers, mais inévitables, « bugs », notamment au moment du chargement d'une sauvegarde.Mais rien de grave, ces incidents n'interfèrent pas sur le bon déroulement de celle-ci. De plus les défauts majeurs d'antan tendent à disparaître pour notre plus grand plaisir. Toutefois, certains subsistent, à l'image du vocabulaire qui n'est pas reconnu, ou des ordres mal interprétés qui peuvent bloquer la partie.Le jeu est prenant et agréable. Il s'agit d'une des meilleures réalisations de Sierra, un excellent jeu d'aventure, où la magie, la sorcellerie, le mystère et le suspense cohabitent avec un certain bonheur. La musique est, comme souvent, très travaillée. Peu plaisante sans interface sonore, elle devient envoûtante avec une carte Ad-Lib, ou équivalente. Cependant, certains passages sont trop répétitifs et d'autres ressemblent de manière suspecte à des morceaux déjà utilisés dans de précédents logiciels (je pense notamment à certains passages de « Space Quest III »). Ouvert à tous, il plaira aussi bien aux plus jeunes qu'aux moins jeunes, un ludiciel pour les 7 à 77 ans en quelque sorte ! Si les légendes des chevaliers de la Table ronde vous font rêver (ou vous ont fait rêver), ou si les films réalisés à partir de ces légendes vous ont plus (« Excalibur » et les Monthy Phyton), ce ludiciel doit enrichir votre logithèque au plus vite.Les disquettes de « Conquests of Camelot » ne sont pas protégées contre la copie, et aucun mot de passe n'est demandé. Le novice ne sera donc pas troublé. Quant à la documentation, elle n'est pas d'une très grande importance pour le joueur émérite, mais peut dépanner les débutants. Le « walkthrough » permet ainsi de bien saisir la philosophie du jeu dès le début — ce qui est nécessaire lorsque l'on n'a pas l'habitude des jeux Sierra.

Stéphane Grosjean

FLASH LOGICIEL

CASTLE MASTER

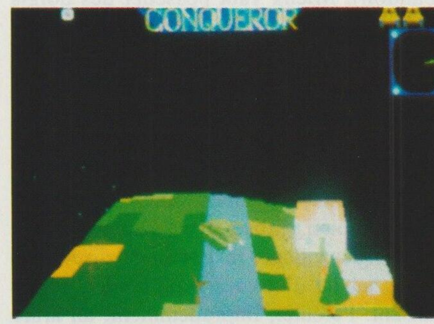
Editeur : Domark
Créateur : Incentive Software
Support : 1 x 3"1/2 et 2 x 5"1/4
Compatibilité : XT, AT, PS/2
Sortie : 06/90
Config. : 256 Ko
Graphismes : H, T, C, E
Type : aventure
Prix : 249 FF
Protégé : disquette
Joueur : 1
Note Globale : 14/20
Graphisme : 14
Son : 6
Animation : 14
Intérêt : 15

Castle Master est une nouvelle aventure d'Incentive Software (Driller, Dark Side, Total Eclipse), reprenant le système d'animation en trois dimensions « freescape ».
Qu'est-ce que le freescape ? Il s'agit d'une gestion en 3D de type vectorielle. Elle permet au joueur de baisser ou de lever la tête — voire de ramper ! —, bref, vous êtes libre de vos mouvements.Au-delà du réalisme de l'action, gérer une telle 3D donne la



possibilité de tendre des pièges plus subtils que dans n'importe quelle autre jeu d'aventure. Il vous faut ainsi faire attention aux éventuels trous dans le sol, chercher des esprits dans les airs, mais aussi des indices ici et là. Ramper permet de découvrir des clés, ou des indices, dissimulés dans des endroits inaccessibles par une autre voie — on prend conscience, par exemple, qu'une table possède bel et bien deux faces ! J'ai plé de clés et d'esprits car ils font partie du jeu. Les clés servent à... ouvrir des portes (étonnant, non ?), les esprits sont maléfiques et susceptibles de causer votre perte. A vous de les trouver avant qu'ils ne soient devenus trop puissants, et de les renvoyer — à coups de pierre — dans l'au-delà, qu'ils n'auraient jamais dû quitter.L'aventure se déroule dans le château de l'Eternité, où votre jumeau est retenu prisonnier par le maître des lieux (le fameux « castle master »). Votre rôle consiste bien évidemment à sauver votre frère, afin de sauver son âme — et la vôtre par la même occasion. Un échec de votre part fera de vous des esprits esclaves, comme vous pouvez en rencontrer dans les couloirs du château. En cours de jeu vous trouverez de la nourriture (fromage, poisson, etc.), source de réconfort et de vitalité. Vous serez également amené à utiliser des trésors, des potions magiques, autant d'éléments qu'il vous faudra découvrir au fil du jeu. Elaborer un plan s'avère nécessaire, car la surface à couvrir (quatre étages plus les catacombes) est importante, sans compter que certains endroits ne sont accessibles que par des trappes depuis l'étage supérieur — et il ne faut pas se tromper de trappe ! Si le scénario n'a rien de bien original, du point de vue de la réalisation le « Castle Master » constitue une réussite. L'animation est même assez fluide, du moins avec un AT. Le système de jeu est des plus simples : quatre touches pour vous diriger, deux pour les mouvements de la tête (haut et bas), et quelques autres pour les options (ramper, marcher, courir). On ne rencontre pas de problème de langue, puisque le jeu peut être chargé en version française. De toute façon, il n'existe pas d'interface syntaxique, et toutes les actions, telles que « ouvrir », « prendre », « utiliser un objet », etc., sont appelées grâce à la même touche. L'effort louable pour simplifier les commandes se ressent toutefois dans le jeu. Certains trouveront « Castle Master » un peu trop rudimentaire.Il s'agit pourtant d'un bon jeu d'aventure, idéal pour débuter. Il comblera même bon nombre de joueurs confirmés. Alors, prêt pour l'aventure ?

S.G.



CONQUEROR

Editeur : Rainbow Arts
Créateur : Jonathan Griffiths
Support : 1 x 5"1/4
Compatibilité : XT, AT
Sortie : 06/90
Config. : 512 Ko
Graphismes : H, T, C, E, V
Type : arcade
Prix : 250 FF
Protégé : non
Joueur : 1 ou 2
Note Globale : 11/20
Graphisme : 13
Son : 6
Animation : 13
Intérêt : 9

Conqueror, jeu d'arcade en trois dimensions, reprend les mêmes graphismes que Virus. C'est-à-dire une vue (assez réduite) en trois dimensions face pleine du terrain. Ici, le poste d'observation est un char d'assaut. Si la vue est amusante, et même agréable au début, elle se révèle assez vite mal appropriée au jeu. Le champ de vision est fortement réduit, et pour peu que vous descendiez une colline, vous ne voyez plus rien ! Une situation gênante, il faut en convenir !
Le manque de réalisme est également flagrant concernant d'autres points, comme le fait que l'on puisse tirer un obus toutes les deux secondes... Les programmeurs viennent de nous inventer le canon à répétition pour les chars d'assauts de la Seconde Guerre mondiale ! Remarquez qu'ils ont aussi trouvé un moyen pour avoir du carburant et des

munitions à l'infini.Autre mauvais point, il est très difficile de juger de la position du canon, en raison de l'unique perspective visuelle proposée. Vous tirez donc souvent au hasard. Ajoutez à cela qu'il est pratiquement impossible de jouer convenablement à Conqueror seul, et vous comprendrez que l'on s'en lasse vite, très vite.Effectivement, pour contrôler votre char et la tourelle il faut utiliser huit touches différentes (« configurables », heureusement), sans compter le tir, et deux autres touches (pause et abandon). Mieux vaut posséder des doigts agiles, et une coordination remarquable. Heureusement, le joueur peut demander à l'ordinateur de s'occuper soit de la tourelle, soit du char. Mais ne comptez plus alors sur un déplacement logique et efficace de l'ensemble tourelle-char. Cependant, Conqueror présente certains points agréables. Ainsi, le paysage est assez bien réalisé, avec des arbres, des rivières, des villages (maisons et églises), des routes, autant d'obstacles qui rendent le jeu plaisant. Percuter, imprudemment, un arbre provoque l'envolée d'un nuage d'oiseaux effarouchés.Regrettons qu'aucune instrumentation ne soit fournie, si ce n'est un vague radar et une carte. Un équipement misérable pour une « simulation ».Conqueror ressemble davantage à un jeu d'arcade qu'à une simulation. D'ailleurs, dès le début du jeu, trois modes différents sont proposés : arcade, semi-arcade et stratégie.La version arcade se limite à un vulgaire duel entre deux chars, le vôtre et celui de l'ordinateur. Si vous gagnez, vous passez au niveau suivant — il y a dix niveaux, numérotés de zéro à neuf. Cette partie de jeu se révèle vite lassante, et vous serez amenés à changer de mode rapidement.Le mode semi-arcade propose une bataille entre vos chars et un adversaire contrôlé par le PC. Cette partie est très simple, il suffit de laisser vos chars —dirigés également par l'ordinateur — aller seuls. En effet, cette « stratégie » vous permettra d'atteindre sans problème le niveau sept !Le mode stratégie est le seul qui semble réellement intéressant. Il permet d'utiliser des grenades et des reconnaissances aériennes pour trouver l'emplacement de l'ennemi, et de situer l'affrontement sur le terrain de votre choix. Des détails insuffisants pour relever le niveau général du jeu. Un achat à envisager avec la plus extrême prudence.

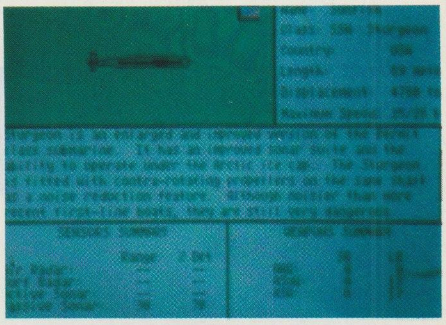
S.G.

FAERYTALE

Editeur : Microillusions
Créateur : David Joiner
Support : 3 x 5"1/4 ou 2 x 3"1/2
Compatibilité : XT, AT ou PS/2
Sortie : 07/90
Config. : 512 Ko
Graphismes : H, T, C, E
Type : aventure
Prix : 300 FF
Protégé : documentation
Joueur : 1
Note Globale : 12/20
Graphisme : 10
Son : 11
Animation : 11
Intérêt : 13

« The Faerytale adventure » vous conduit à la recherche d'un talisman. Le talisman est une petite sculpture d'un taureau, qui a été dérobée dans votre village, et qui possède de très grands pouvoirs. Privé de son influence, votre village court à sa perte.Il y a de cela peu de temps, une nuit, un étrange événement s'est produit : un spectre noir est apparu au milieu du bourg, et derrière lui, se tenait une armée de morts et de squelettes ! Ils étaient armés d'épées et de masses d'armes, et se sont livrés à d'étranges rites jusqu'au petit matin. C'est le lendemain que la population découvrit que le talisman avait disparu. N'étant plus protégé, voici votre village à la merci de n'importe quel envahisseur, ogres ou barbares des montagnes, attirés vers le sud à la suite d'un terrible hiver. Devant une telle situation, deux choix se présentaient : demander de l'aide au roi, ou au chevalier rouge. Hélas, vous vous en doutiez, leurs emplois du temps respectifs étaient des plus chargés, et ils ne purent se libérer. A vous donc de prendre les choses en main, et de tenter de soustraire le pays à l'influence maléfique du « nécromancien ».Voici, tracé à gros traits, le scénario de cette aventure. S'il reste classique pour une histoire fantastique, il faut noter qu'il est tout de même bien conçu. Tous les ingrédients d'un bon jeu d'aventure sont réunis. D'où une légitime déception après quelques parties.Les graphismes sont affreux, les personnages grossiers (graphiquement, évidemment), la représentation du terrain est pauvre. De plus, votre personnage se déplace avec une certaine disgrâce. Pour arriver à vos fins, il vous faudra parcourir l'ensemble du territoire géré— et Dieu qu'il est grand ! D'après les programmeurs, il se compose de 144 écrans en longueur sur 100 de largeur ! De quoi vous occuper pendant un petit bout de temps.Pour ce qui est du contrôle du personnage (il s'appelle Julian), vous pouvez utiliser soit la souris, soit votre joystick préféré, ou plus simplement le clavier. Le déplacement s'effectue dans huit directions (haut, bas, droite, gauche et les diagonales). Au cours du jeu, vous devrez combattre bon nombre d'adversaires — généralement plus faibles que vous -, une manière d'augmenter vos points de bravoure. En fouillant vos adversaires, vous trouverez de nombreux objets, comme de l'argent, des armes, des pierres, etc. L'argent est un élément important puisqu'il permet d'acquérir de la nourriture et des ustensiles divers. Au départ, Julian possède un certain nombre de points de vie, de bravoure, de bonté et de chance. La chance décidera que vous serez ou non, ramené à la vie par une gentille fée, pleine d'attentions pour vous. La bonté servira à communiquer avec les quelques personnes, assez rares, qu'il ne vous faudra pas tuer.Le maniement est assez simple. Pour le combat vous n'utilisez qu'une touche, mais il faut cependant faire face à l'adversaire. La magie tient une place de choix, et vous pourrez trouver sept objets magiques différents, « jetables », soit sept touches différentes supplémentaires. Dans le souci d'éviter des massacres inutiles, il existe également une option pour communiquer - avec trois touches pour parler, questionner et crier. Le reste des commandes semble un peu trop compliqué. Rien que pour acheter, pas moins de sept touches sont accessibles. C'est beaucoup. Notons qu'une option de sauvegarde permet de conserver les parties engagées. L'intrigue est intéressante, mais la réalisation n'est pas à la hauteur, et le jeu en pâtit. Cependant, un effort a porté sur l'environnement musical.Destiné en priorité aux fervents des aventures fantastiques, Fearytale ne saurait être conseillé aux débutant souhaitant s'initier au genre.

S. G.



HARPOON

Editeur : Mirrorsoft
Créateur : Three-Sixty
Support : 3 x 5"1/4
Compatibilité : XT, AT
Sortie : 06/90
Type : guerre
Prix :
Protégé : non
Joueur : 1
Config. : 640 Ko, disque dur
Graphismes : H, T, C, E
Note Globale : 17/20
Graphisme : 12

MICRO impression



VIDEO PAINT

DESCRIBE

TRUEFORM

LEXIQUE DE LA TYPOGRAPHIE VIII

M 1201 - 13 - 20,00 F-RD

N° 13 AOUT- SEPT 1990

Son : 06
Animation : -
Intérêt : 17

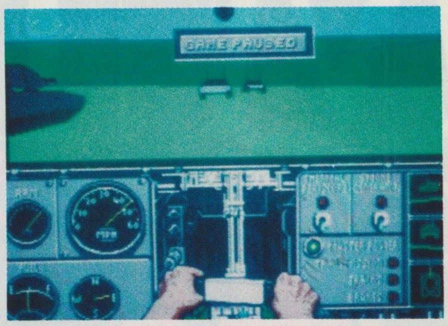
Harpoon est une simulation de combats navals dans la mer de Norvège, opposant l'O.T.A.N. à l'U.R.S.S.
En fait de simulation, Harpoon tient davantage d'un wargame. Vous ne vous trouvez pas aux commandes d'un unique navire, mais vous contrôlez une, voire plusieurs, flottilles. Vous voici donc, en quelque sorte, à la place de l'état-major.
Mais, rassurez-vous, il n'est pas nécessaire d'avoir des connaissances en matière de wargame pour apprécier ce jeu à sa juste valeur. Et quelle valeur ! La réalisation est des plus soignées, utilisant un environnement du type Windows, avec menus déroulants et multifenêtrage.Les menus déroulants permettent un accès aisé aux nombreuses options du jeu, le réglage de la vitesse, du cap, la formation ou la durée des tours de jeu, etc. Le multifenêtrage permet, quant à lui, d'avoir une bonne vision de la situation. D'abord, en haut à gauche, s'affiche une petite carte de l'aire de jeu, dans laquelle vous déplacez un rectangle. La zone définie par le rectangle est représentée dans une grande fenêtre, en haut à droite, avec un zoom à quatre niveaux. Dans cette zone, vous pouvez sélectionner la flottille qui vous intéresse. Elle sera alors reproduite dans une troisième fenêtre, plus petite, en bas à gauche de l'écran. Elle vous donnera une meilleure appréciation de la formation de la flottille, avec là encore un système de zoom à quatre niveaux. La quatrième fenêtre fournit des renseignements sur le navire sélectionné dans la fenêtre précédente.
En dépit des nombreuses options, le jeu reste assez facile d'accès, grâce à une prise en charge des défenses par l'ordinateur. De plus, une grande liberté de jeu est offerte. Toutes les données de chaque navire (cap, vitesse, etc.) peuvent être définies, ainsi que la durée d'un tour de jeu (d'une seconde pour une seconde jusqu'à une seconde pour une demi-heure). Il faut noter la formidable bibliothèque livrée avec le jeu, qui fournit un grand nombre de données sur chaque navire ou appareil du jeu.
Au début de chaque partie, il faut choisir son camp : OTAN ou URSS. Quant aux niveaux de jeu, ceux-ci sont bien échelonnés sur treize scénarios, du plus simple, permettant de s'habituer à l'environnement, au plus compliqué, qui fait entrer en jeu plusieurs flottilles amies et ennemies. Mais au-delà d'un simple wargame naval, Harpoon permet de gérer aussi les forces aériennes présentes sur certains navires, dans certains scénarios du moins. L'option de sauvegarde permet d'interrompre momentanément la partie en cours, qui peut s'avérer très longue avec les scénarios les plus évolués, pour la reprendre plus tard. Harpoon est accompagné d'un bon manuel, complet, décrivant bien toutes les options du jeu. Seule réserve, il est écrit en anglais.Pour bien apprécier Harpoon, il convient de posséder un matériel assez puissant - AT, souris, mémoire vive d'un Mo, disque dur, écran EGA -, ce qui représente un inconvénient majeur.Des configurations moindres font apparaître des limites : la souris est impérative afin d'assurer une souplesse satisfaisante dans l'environnement, et un Mo de mémoire vive est nécessaire pour utiliser l'ensemble des scénarios disponibles.
Si vous possédez le matériel adéquat, et que les wargames vous passionne, alors pas d'hésitation, courez vite acheter Harpoon !

S.G.

TANK

Editeur : Mirrorsoft
Créateur : Spectrum Holobyte
Support : 1 x 5"1/4
Compatibilité : AT, PS/2
Sortie : 06/90
Config. : 640 Ko
Graphismes : E, V
Type : Simulation de tanks
Prix : 250 FF
Protégé : non
Joueur : 1 (ou plus si modem)
Note Globale : 14/20
Graphismes : 12
Son : 7
Animation : 16
Intérêt : 13

Tank est une nouvelle simulation de char d'assaut en trois dimensions vectorielles, assurant un réalisme acceptable.Il s'agit, une fois encore, de contrôler un modèle du célèbre char américain, le M1 Abrams. Mais cette fois, les programmeurs ont tenté, avec un certain succès, de rendre le jeu plus complet que les précédents, pour éviter qu'il ne soit qu'une copie. Ainsi, Tank est davantage qu'une simulation, c'est aussi un jeu alliant stratégie et wargame. Vous pouvez effectivement n'utiliser qu'un char, avec toutes les options habituelles, ou préférer diriger jusqu'à seize chars. C'est là qu'intervient la notion de wargame, puisque vous contrôlez l'évolution des chars d'après une carte. L'aspect stratégique est certes toujours présent, mais il prend ici une place accrue grâce à la possibilité d'utiliser des appuis aériens, une aide de l'artillerie, ou des reconnaissances aériennes. Tank demeure cependant, et avant tout, une (bonne) simulation. Le jeu offre de nombreuses vues (assez habituelles pour ce genre de logiciel). Vous pouvez vous placer dans le rôle de



l'artilleur, du conducteur, du commandant du char, mais aussi dans ceux du mitrailleur ou d'un observateur placé sur la tourelle. Une foule de petits détails concourent au réalisme de Tank, à l'exemple des douilles qui s'éjectent lors de l'utilisation de la mitrailleuse. Parmi les multiples angles de vue envisageables, notons celui d'un observateur fixe se trouvant devant le char, ou encore la vision du char depuis l'extérieur, de derrière ou de côté. Les difficultés sont savamment dosées.

Tank propose quinze missions classées par ordre croissant de difficulté. Elles se situent dans trois lieux différents. Dont Fort Knox, lieu d'entraînement et base de vos engins. Les cinq missions proposées sont simples et vous permettent de vous initier aux commandes du jeu. L'Europe centrale, avec cinq autres missions déjà plus difficiles, et le Moyen-Orient, abritent le reste des aventures.A chacune des missions, le niveau de jeu (trois niveaux) peut être configuré. Les aspects stratégique et wargame du jeu ne le rendent cependant pas trop compliqué. En effet, toutes les commandes ont été regroupées dans différents répertoires logiques, et se sont vues fortement simplifiées. Simplifiées certes, mais toujours très complètes. Ainsi, vous pouvez définir les mouvements d'un char ou d'un groupe de chars et leur donner des ordres (défendre, attaquer, etc.) Vous contrôlez aussi leur vitesse, ainsi que le type de formation (huit différentes). Une partie de la carte de champ de bataille peut être agrandie grâce à un système de zoom à seize niveaux. Vous pouvez également demander de l'aide sous forme de soutien aérien, de tir d'artillerie, ou de reconnaissance en hélicoptère. Ce jeu, très complet, reste simple à mettre en oeuvre. Ce qui ne gâte rien, il possède une bonne documentation rédigée en français. Une franche réussite qui comblera les passionnés de simulations de guerre.

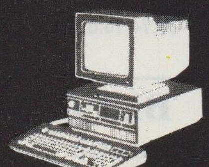
S.G.

LIBRE OU OCCUPÉ PASSEZ AU SALON

RFM LA RADIO FM 103.9



2ème SALON DE LA MICRO



MINITEL 36-15
code CFDL

LA MICRO VRAIMENT INDIVIDUELLE

26-29 OCTOBRE 1990

ESPACE NORD - LA VILLETTE

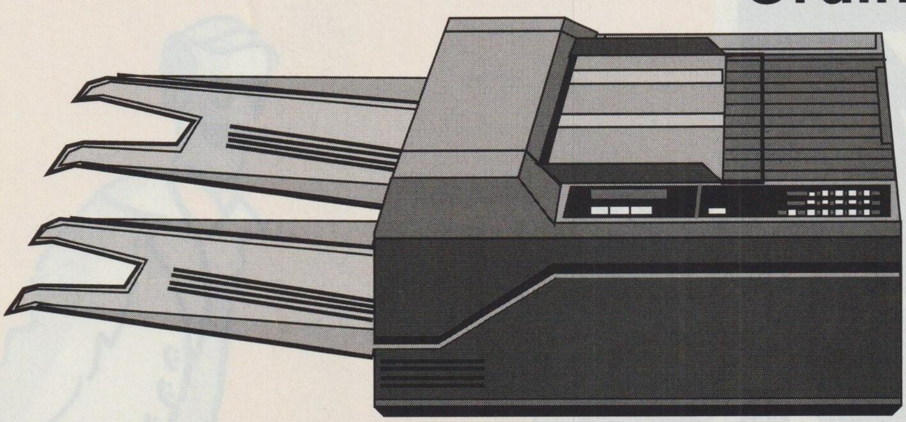
la Villette — la grande halle

Boum & Bang RCS PARIS B 350 063 822

Cerclez info pub 14

POUR S'AFFRANCHIR DE LA PHOTOCOMPO

STATION PAO POSTSCRIPT 600 DPI



Ordinateur AT 386 SX

1 lecteur 5' 1/4 1.2Mo et 3'1/2 1.44Mo

8 Mo Ram Disque dur 80 Mo

Ecran monochrome haute résolution

Ms-Dos 4.01 / Windows 3.0

Imprimante 600 dpi

Carte DPtek 600 dpi

UltraScript

Imprimante Canon 4p/mn

54900 F HT

Option

Scanner 600 dpi / 256 niv.gris

14900 F HT

En démonstration sur rendez vous.
Vérifiez bien que le Micro Video de votre région dispose bien de ce produit.

La disponibilité peut varier suivant les magasins

Dans la limite des stocks disponibles

Prix différents en Belgique, renseignez vous !

MICRO VIDEO

la passion d'un spécialiste, la puissance d'une chaine.

PARIS	BORDEAUX	DAX	LYON
8, rue de Valenciennes 75010 Paris ☎ 40.37.92.75 / 40.34.97.80 + **Ouvert du Mardi au Samedi de 10H à 19 H** *Métro: Gare de l'Est / Gare du Nord*	3, cours Alsace et Lorraine 33000 Bordeaux ☎ 56.44.47.70	56, Avenue Victor Hugo 40100 Dax ☎ 58.74.18.63	11,12 cours Aristide Briand 69300 Caluire ☎ 72.27.14.74

NOUVEAU ! MONTPELLIER	NOUVEAU ! NANTES	PERPIGNAN	TOURS	BELGIQUE – BRUXELLES	BELGIQUE – DINANT
7 rue Raoux 34000 Montpellier ☎ 67.58.39.20	6 rue Mazagran 44000 Nantes ☎ 40 69 15 92	8, avenue de Grande Bretagne 66000 Perpignan ☎ 68 34 24 40	81, rue Michelet 37000 Tours ☎ 47.05.78.50	1, rue Dons 1050 Bruxelles ☎ 02 / 648 9074	21 place communale 5198 ANHEE ☎ 082 / 611451

Service lecteur étendu...

L'INFORMATION EN DIRECT

Si vous souhaitez obtenir des renseignements plus complets sur des produits cités, complétez la carte « service lecteur » qui se trouve page 101, en vous référant aux numéros donnés ici.

Le système d'exploitation OS/2 1.2	p 42 à 71	service lecteur 170
Le calculateur scientifique HP-48 SX	p 76 à 88	service lecteur 169
Le langage Turbo Forth	p 112 à 115	service lecteur 168
Le PS/1 d'IBM	p 122 à 126	service lecteur 127
Le livre « Découvrez Turbo Pascal sur PC »	p 126 à 135	service lecteur 126
Le livre « Turbo Pascal version 4 »	p 126 à 135	service lecteur 125
Le llivre « Programmer en Turbo Pascal versions 3 et 4 »	p 126 et 135	service lecteur 167
Le livre « Exercices en Turbo Pascal »	p 126 à 135	service lecteur 166
Le livre « Programmer en Turbo Pascal versions 4 et 5 »	p 126 et 135	service lecteur 124
Le livre « Turbo Pascal par la pratique »	p 126 et 135	service lecteur 165
Le livre « Maîtriser Turbo Pascal 5.5 et la POO »	p 126 et 135	service lecteur 164
Le livre « Turbo Pascal, bibliothèques de procédures »	p 126 et 135	service lecteur 163
Le livre « Approche de la programmation ...»	p 126 à 135	service lecteur 162
Le livre « Bibliothèque de programmes en Turbo Pascal »	p 126 à 135	service lecteur 161
Le livre « Le Grand Livre de Turbo Pascal, 5.0 et 5.5 »	p 126 à 135	service lecteur 160
Le livre « Turbo Pascal, versions 5.0 et 5.5 »	p 126 à 135	service lecteur 159
Le livre « Le Livre de Turbo Pascal sur PC (version 4) »	p 126 à 135	service lecteur 158
Le livre « Au coeur de Turbo Pascal et le désassembleur »	p 126 à 135	service lecteur 157
Le livre « La bible de Turbo Pascal 5.0 et 5.5 »	p 126 à 135	service lecteur 156
Le livre « Programmation avancée en Turbo Pascal »	p 126 à 135	service lecteur 155
Le livre « Turbo Pascal 5.5 et la programmation objet »	p 126 à 135	service lecteur 154
Le livre « Turbo Pascal orienté objet »	p 126 à 135	service lecteur 153
Le livre « Data Structured using Pascal »	p 126 à 135	service lecteur 152
Le livre « B-Arbre et sequentiel indexé»	p 126 à 135	service lecteur 151
Le livre « Pilotez votre imprimante en Turbo Pascal »	p 126 à 135	service lecteur 150
Le livre « Pascal étendu, projet de norme Ieee, Iso, Ansi »	p 126 à 135	service lecteur 149
Le livre « Turbo Pascal et ses fichiers, versions 4 et 5»	p 126 à 135	service lecteur 148
Le livre « Langage Pascal et logique de premier ordre»	p 126 à 135	service lecteur 147
Le livre « Graphisme avec Turbo Pascal et le Graphic ...»	p 126 à 135	service lecteur 146
Le livre « Apprendre Pascal et la récurssivité... »	p 126 à 135	service lecteur 145
Le jeu « Conquest of Camelot »	p 126 à 135	service lecteur 144
Le jeu « Conqueror »	p 160 à 164	service lecteur 143
Le jeu « Tank »	p 160 à 164	service lecteur 142
Le jeu « Castle Master »	p 160 à 164	service lecteur 141
Le jeu « The Faerytale adventure Book »	p 160 à 164	service lecteur 140

INDEX DES ANNONCEURS

Adalog	p 67	info pub 22
Alif	p 6 et 7	info pub 3
Alco Systèmes	p 23	info pub 10
Asia Star Computer	p 4 et 5	info pub 2
Asia Star Computer	p 19	info pub 8
Alliance	p 129	info pub 38
Abacore	p 129	info pub 31
C.P.I.O.	p 49	info pub 18
Côte Ouest	p 15	info pub 6
Davel Trading	p 89	info pub 25
Delphine Software	p 161	info pub 34
Enosys	p 121	info pub 29
H.A.	p 13	info pub 5
Informatique Services	p 8 et 9	info pub 4
KADDE	p 27	info pub 11
Komelec	p 81	info pub 24
La Dédicace	p 33	info pub 13
Intel Business Micro	p 167	info pub 36
Logi PC	p 109	info pub 27
Logi PC	p 111	info pub 28
Logidata	p 133	info pub 32
KADDE	p 27	info pub 11
Maine Informatique	p 53	info pub 19
Micro Application	p 17	info pub 7
Micro Video	p 45	info pub 17
N.A.I.T. Informatique	p 55	info pub 20
Partner Concept	p 75	info pub 23
Polywell	p 21	info pub 9
Log'Up	p 121	info pub 30
R.C.I.	p 94	info pub 26
Salon de la Micro	p 37	info pub 14
Sicob	p 2	info pub 1
Olitec	p 168	info pub 37
Softworld	p 63	info pub 21
Serelec	p 135	info pub 33
Vidéoshop	p 41	info pub 16
Vidéoshop	p 39	info pub 15

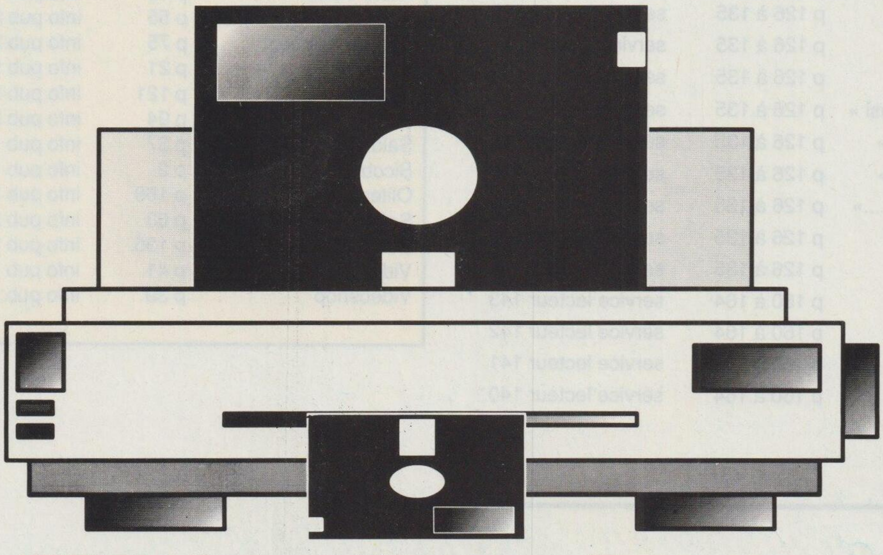
Le prochain numéro...

Le prochain numéro d'*Action PC* sera dans tous les points de vente habituels début décembre.

Au sommaire de ce numéro 7 – spécial portables – vous trouverez notamment :

- la communication (2e partie)
- les portables légers
- DR-DOS 5.0

et toutes nos rubriques habituelles...



ACTION PC améliore la productivité des ordinateurs PC

ACTION PC

PUBLICATION MENSUELLE
11 NUMÉROS PAR AN
ÉDITÉE PAR PRESSIMAGE
ISSN : 0997-0088
COMMISSION PARITAIRE : 71 681
DÉPOT LÉGAL : A PARUTION

RÉDACTION - ABONNEMENTS
19, RUE HÉGÉSIPPE-MOREAU
75018 PARIS
(1) 45.22.38.60

PRESSIMAGE
S.A.R.L. AU CAPITAL DE 2000 F
CODE APE : 5120
210, RUE DU FBG ST MARTIN
75010 PARIS

ACTION PC EST DISTRIBUÉ
PAR LES N.M.P.P.

RÉDACTION

RÉDACTEUR EN CHEF
ÉRIC VON ASCHEBERG

CONSEILLER A LA RÉDACTION
JEAN-FRANÇOIS BALAINE

REDACTEUR
GUILLAUME DE BREBISSON

CONSEILLERS TECHNIQUES
DANIEL MARTIN
FRANÇOIS PIETTE

SECRÉTAIRE DE RÉDACTION
FRANÇOISE GERMAIN

ONT COLLABORÉ A CE NUMÉRO
PASCAL FIEVET
CHRISTOPHE GROSJEAN
STEPHANE GROSJEAN
DANIEL MARTIN
VINCENT POUILLEY
JEAN-MICHEL SPINER
CHRISTIAN TAVERNIER

COUVERTURE ET ILLUSTRATIONS
JAMEL TAYEB

MAQUETTE ET FABRICATION
MICHEL LHOPITAULT
OLIVIER FERFACHE
ANNABELLE TBIRIAN
VITTORIO CARAMAZZA
CALH HUNG TRAN

PUBLICITÉ

DIRECTEUR DE LA PUBLICITÉ
ALBERT ELBAZ
TEL. 16 (1) 43 87 01 39

DIRECTEUR DE LA PUBLICATION
GODEFROY GIUDICELLI

Imprimé en France par SIMA TORCY
Z.I. de Torcy, 77200 Torcy

INTEL BUSINESS MICRO COMPUTER SYSTEMS
US QUALITY

SUPER VGA couleur
Moniteur 14 pouces

HDD 40 Mo, 28 ms
Floppy 1,2 ou 1,44 Mo
1 // + 2 series
DOS 3.3 + windows 3
10 Mo de shareware
Souris + Pilote
Gestionnaire HDD
Emplacement Co-PRo

216
i286–16 Mz
Ram 2 Mo → 8 Mo
Landmark 21 Mz
16990 HT

320-325-333
i386–20/25/33 Mz
Ram 4 Mo → 16 Mo
Landmark 29 Mz, 42 Mz, 55 Mz

20 Mz **20990 HT**
25 Mz **27990 HT**
33 Mz **33990 HT**

320 SX
i386–20 Mz
Ram 3 Mo → 8 Mo
Landmark 24 Mz
18990 HT

425
i486–25 Mz
Ram 4 Mo → 16 Mo
Landmark 117 Mz
58990 HT

OPTIONS

• Disque durs :	80 Mo	3990 HT
	115 Mo	5590 HT
• 19 " VGA couleur		5590 HT

TOUS NOS SYSTEMES SONT PRETS A L'EMPLOI, configurés et testés en usine 144 heures.
Les composants sont de 1re qualité, ce qui nous permet de garantir nos machines 2 ans.
Vous avez 30 jours pour tester votre système : SATISFAIT OU REMBOURSE.
MAINTENANCE SUR SITE PENDANT UN AN INCLUS.

12000 REFERENCES EN STOCK
EN PROMOTION

Co-processeurs Intel ou CYRIX Fasmath 4 fois plus rapide :

80387-SX.....**2790 HT**
80387-16.....**3090 HT**
80387-20.....**3190 HT**
80387-25.....**4290 HT**
80387-33.....**5290 HT**

Télécopie - Modems

SAMANTHA FAX pour PC/AT
100 % Compatible Hayes émulation Minitel
Télécopie Agréé PTT **6990 HT**

Imprimantes

PANASONIC KX-P1124 **3390 HT**
Laser Kx-P4420................................ **11500 HT**

Disques durs

40 Mo, 28 ms **2390 HT**

Scanner

HANDSCAN **1890 HT**
PERSONAL A4 **5990 HT**

Tablettes à digitaliser

CAO/FAO/DAO A3 **6990 HT**

SOFTS

MS-DOS 4.01 Windows 3
Tous les logiciels pour Windows 3

Réseaux

Carte ARCNET.......**890 HT**
Carte ETHERNET **1290 HT**
MAINLAN pc-pc..**5990 HT**

Divers

1 Mo simm **890 HT**

Cerclez info pub 36

E
CLOWN'S TAIWANAIS

386-SX • 16	1 Mo Ram Série ∈ //	286
Ecran 14" + carte MGP	Disque Dur 20 Mo	Ecran 14" bif, Ambre ou B/P
9590 F HT	Clavier	**8250 F HT**

Possibilité de financement de 4 à 84 mois (nous consulter). Votre système en crédit sur 3 mensualités.
Paiement Carte Bleue, chèque, contre remboursement
Commande par téléphone, télécopie ou courrier. Disponibilité sous 24 h (Paris), 72 h (province).

RECHERCHONS REVENDEURS, COMMERCIAUX ET V.R.P.

2, PLACE DE LA DÉFENSE BP 240 - 92053 PARIS LA DÉFENSE
TÉL : (1) 46 92 24 24 • FAX : (1) 46 92 24 00


Devant le succés de l'opération
OLITEC
joue les prolongations
5ème anniversaire
OLITEC
exceptionnel !
1990 F HT
1 carte modem
PS ou PC 2400 - Hayes
(V 21, V 22, V 22 bis, V 23)
Paul Rousseau Communication
Pour tous renseignements :
Tel 83 21 95 15 • Fax 83 29 00 57
2360,14 TTC (Prix de vente conseillé)
Offre valable du 1/10/90 au 31/10/90,
limitée aux 500 premières
cartes commandées
L'INTELLIGENCE SERVICE
OLITEC, CONCEPTEUR ET FABRICANT FRANÇAIS
DE MODEMS ET DE CARTES MODEM
OLITEC • BP 592 • 54009 Nancy Cédex • Serveur 83 29 56 67
Cerclez info pub 37